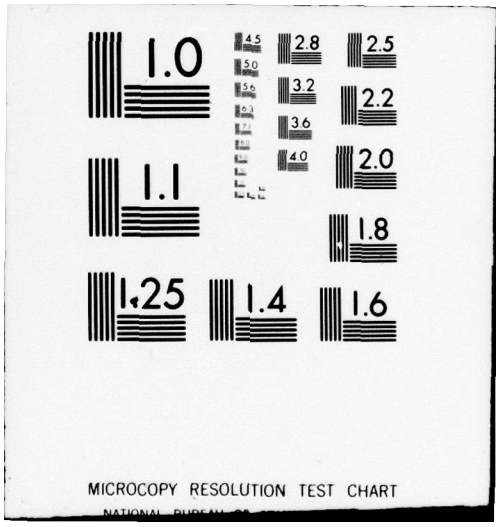
AD-A074 435 HONEYWELL INFORMATION SYSTEMS INC MCLEAN VA F/G 5/2
STATISTICAL COLLECTION FILE (SCF) MAINTENANCE MANUAL,(U)
NOV 77 D BEILFUSS, J BIELSKI, R DAY, R EWING DCA100-73-C-0055
UNCLASSIFIED NL

1 OF 3
ADA 074435


1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
4.5
5.0
MICROCOPY RESOLUTION TEST CHART

DATA PROCESSING SHEET

PHOTOGRAPH THIS SHEET

LEVEL III

INVENTORY I

DDC ACCESSION NUMBER

AD A074435

"SCF [redacted] Maintenance Manual"

DOCUMENT IDENTIFICATION

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

DISTRIBUTION STATEMENT

Accession For
NTIS GRA&I
DDC TAB
Unannounced
Justification
By
Distribution/
Availability Codes
Dist | Avail and/or special
A

DISTRIBUTION STAMP

DDC RECEIVED SEP 25 1979 D

DATE ACCESSIONED

79 09 19 016

DATE RECEIVED IN DDC

PHOTOGRAPH THIS SHEET

AND RETURN TO DDA-2

AD A074435

CCTC Tech Library
11440 Isaac Newton Sq
Reston VA 22090

Honeywell

STATISTICAL COLLECTION FILE (SCF)
MAINTENANCE MANUAL
TASK 626 SUBTASK 2
NOVEMBER 21, 1977

WWMCCS

Honeywell

CCTC TASKING STATEMENT 626
SUBTASK 2

STATISTICAL COLLECTION FILE (SCF) MAINTENANCE MANUAL

Prepared For: Defense Communications Agency
Command and Control Technical Center

Contract Number: DCA100-73-C-0055

Contractor: Honeywell Information Systems, Inc.

Prepared By: Don Beilfuss
John Bielski
Richard Day
Ronald Ewing
Thomas Hunzeker
David Kayden
Steven Robbins

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Statistical Collection File (SCF) Maintenance Manual		5. TYPE OF REPORT & PERIOD COVERED Maintenance Manual 6. PERFORMING ORG. REPORT NUMBER N/A
7. AUTHOR(s) Don Beilfuss, Ron Ewing, John Bielski, Thomas Hunzeker, Steve Robbins, Richard Day and David Kayden		8. CONTRACT OR GRANT NUMBER(s) DCA100-73-C-0055
9. PERFORMING ORGANIZATION NAME AND ADDRESS Honeywell Information Systems, Inc. 7900 Westpark Drive McLean, Virginia 22101		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS Task 626-Subtask 2
11. CONTROLLING OFFICE NAME AND ADDRESS Honeywell Information Systems, Inc. 7900 Westpark Drive McLean, Virginia 22101		12. REPORT DATE 11/21/77 13. NUMBER OF PAGES 280
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office) Defense Communications Agency Joint Technical Support Activity Reston, Virginia		15. SECURITY CLASS. (of this report) Unclassified 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)
This document is released for unlimited distribution.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)
N/A

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)
Operating System
General SCF Expansion Program
GCOS
Statistical Collection File (SCF)

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)
This document contains the information necessary to maintain the changes which were made to GCOS during the implementation enhancements to the Statistical Collection File.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

Table of Contents

Page

SECTION 1 GENERAL OVERVIEW..... 1

Section 1 contains an overview of both WW6.4 and WW7.1 SCF enhancements. It provides the reader with an in-depth overview of this WW6.4 implementation in light of the total design.

SECTION 2 SCF RECORDS..... 28

Each record is described in this section. Information includes the record function and GCOS modules modified to affect each function. The records are presented in numerical order.

SECTION 3 GCOS MODULES..... 69

Each GCOS module which was altered is listed alphabetically in this section. The section consists of module functions and SCF records created by each module.

SECTION 4 .MSCF HIERARCHICAL FUNCTION DIAGRAM....186

This section consists of the Hierarchical Function Diagram for .MSCF.

SECTION 5 .MPAAF HIERARCHICAL FUNCTION DIAGRAM...238

SECTION 6 CONSOLE VERBS HIERARCHICAL FUNCTION DIAGRAM.....249

SECTION 1

GENERAL OVERVIEW

1.1 SYSTEM DESCRIPTION

The Statistical Collection File is being enhanced in two phases. Phase I will be implemented in release 6.4 while phase II will be available in release 7.1. An overview of the Statistical Collection File enhancements is presented in the following paragraphs. This complete description will indicate the records and parts of records which are applicable to each release. The reader should be aware that this maintenance document is to be referenced for the WW6.4 release only.

New system functions to be implemented in release WW7.1 require the definition of additional SCF records to perform various accounting/audit functions related to these new elements. Because of the large number of records defined a scheme has been developed to categorize SCF records by record class. The scheme allows for the definition of up to 350 SCF record types in ten classes. In addition, allowance has been made for the addition and control of user-defined SCF records.

The SCF classes are defined as follows:

Class	Record Type Range
Batch	001 - 035
Terminal Session	101 - 135
Security	201 - 235
System Files	301 - 335
System Instrumentation	401 - 435
Process Instrumentation	501 - 535
User	601 - 635
User	701 - 735
User	801 - 835
User	901 - 935

The first digit of the record type represents the class and the remaining two digits are the record number (01-35) within a class.

The following discussion describes the interaction of SCF records with GCOS and is segmented by SCF record classes. This overview is quite general and is not meant to describe GCOS and its interfaces in detail.

1.1.1 STANDARDIZATION OF RECORD HEADER

A standard record header is a means of providing consistency among the SCF records. The standard header contains the following information: system identification (both User Host and Server Host), job SNUMB, unique job identifier, time of

record creation, USERID, program number, activity number, and terminal identification (DATANET number, terminal ID, and terminal type). The user host and server host system identifications are for use with the Prototype WWMCCS Intercomputer Network (PWIN). The unique job identifier is generated for each job processed and provides a means of tracking jobs with duplicate SNUMBs.

The standard header is a part of each and every SCF record. If some of the header information is inappropriate for a particular record, the particular field is not omitted but simply not used.

1.1.2 BATCH SCF ACCOUNTING

During the course of a batch job through the system, there are seven phases of system function:

- System Input
- System Scheduling
- Peripheral Allocation
- Memory Allocation
- Execution
- Termination
- System Output

The following describes each of these phases and identifies the points from which SCF records are generated.

1.1.2.1 System Input

A job can be entered into the system from three sources: local card reader, Input Media Conversion (IMCV) tape, or shared mass storage. The shared mass storage jobs originate as a result of a console operator spawn, a user issuing a MME GENEWS, a TSS spawn, or remote batch input. As each job is processed, GEIN makes the first attempt at possible deletion of the job for any number of reasons (invalid user identification, incorrect control cards, etc.). GEIN generates the J* and *J files. The *J file contains the data subfiles for the job. The J* file contains the control cards and pointers to the data files on *J. Just prior to writing block 0 of J*, GEIN classifies the job to either the express, hold, or normal queue. If there is a site-written .MSCAN module present, the .MSCAN module then provides any non-standard priorities and/or classifications. The information utilized by the .MSCAN module consists of type of resources required, the type of activities defined, the amount of core required, and the amount of processor time required.

At the time the job is passed to the System Scheduler, a type 001 SCF record is written. This record contains the job start and stop time, the record count, memory size, processor time, IOM time for the input device, *J, and J*, the job point of origin (CR RDR, IMCV, REMOTE, OPSPWN, or SNUMB of spawning job), and the .MSCAN module parameters (resources required, activity definitions, System Scheduler class priority), the Security Classification codes (SCCs), and the first $IDENT card image.

If GEIN deletes the job, the type 001 record contains a deletion code and a type 016 SCF record is written. This record contains both the deletion code and the deletion message.

N-Level Priority Processing (7.1 only) is a means of isolating priority jobs within each phase of job flow and servicing their demands before servicing lower priority jobs. The priority structure supports the following five levels:

- ROUTINE
- PRIORITY
- IMMEDIATE
- FLASH
- FLASH OVERRIDE

Any priority level other than ROUTINE is indicated on a $ PRIOR control card. When this card is processed with GEIN, an SCF record type 002 is written to record the priority level.

1.1.2.2 System Scheduler

Jobs processed by GEIN are passed to the System Scheduler for processing and scheduling according to the assigned class and priority. Dependent upon the number of jobs in the system, the relative priorities, and other scheduling priorities, the job is either cataloged for later execution or passed directly to the Peripheral Allocator. If during processing the job is deleted because of bad J* file, operator delete, or incorrect $MSG3 cards, an SCF record type 016 records the delete message. When the job is passed to the Peripheral Allocator, the System Scheduler assigns a program number to the job, makes an entry in the Peripheral Allocator input queue, and enables the Peripheral Allocator via a Dispatcher entry.

1.1.2.3 Peripheral Allocator

The Peripheral Allocator begins its processing by reading the J* file and producing an encoded and shortened version

of it. The Peripheral Allocator determines if the job is constructed correctly and whether it can be run; if not, the job is deleted. If the job is deleted by PALC, a type 016 SCF record will record the delete message.

Allocation proceeds on a one-activity-at-a-time basis. The first time an activity is considered for allocation, the exact memory and peripheral requirements are extracted from the J* file. A peripheral summary is then created from which the Peripheral Allocator can perform a gross resource check. Peripheral allocation is accomplished by matching the job requirements with the available resources. When peripheral allocation is successfully completed, the Slave Service Area (SSA) image (including Peripheral Allocation Tables (PATs) for files, SYSOUT lines, etc.) is written to *J, and an entry is made in the Memory Allocator input queue. Also, at this time a type 003 SCF record is written. This record contains, for each device allocated, the file code, disposition, device type, IOM number, channel number, device number, and device-related information such as llink position, initial llink size, maximum llink size for mass storage and the reel number for tapes.

In conjunction with peripheral allocation, FMS records are generated. As a result of allocator-FMS interface, the FMS may perform functions of file creation, deletion, allocation, and accounting. SCF record type 006 accounts for file creation, file deletion, and file-size increase. Record type 006 contains the SMC name, FMS function code, the space and time product, total resources available, and resources used. SCF record type 005 accounts for the creation, modification, or deletion of a catalog or file description, or the setting/resetting of abort or security locks. SCF record type 004 accounts for file allocation attempts when the password is incorrect or the requested action is not permitted. SCF record type 012 accounts for file deallocation attempts when the password is incorrect or the requested action is not permitted. Record types 005, 004, and 012 include the FMS function code, the diagnostic message if the request was not completed successfully, the device SCT address and seek address if the file is secure, the catalog/file string, permissions allowed/requested, and flags to indicate file status.

1.1.2.4 Memory Allocator

The primary purpose of the Memory Allocator is control and management of memory for both user jobs and system modules. For the initial allocation, the required space is located, the area is cleared to zero, and a bootstrap module is loaded into the allocated memory. The bootstrap program then reads the SSA image in, sets up the job start message, and

loads the proper software. The Memory Allocator enables the job via a call to the Dispatcher, and the job is a candidate for execution.

1.1.2.5 Execution

During execution, each activity and event is under the supervision of the Dispatcher, providing service requests, exchange of information and control functions. While in execution, the job may release or request additional peripherals or memory. To request either additional peripherals or memory, a MME GEMORE is processed. When the MME GEMORE is successful, an SCF record type 010 is written. For the allocation of additional peripherals, the same device information as reported in SCF record type 003 is reported. For the allocation of additional memory, the SCF record type 010 records the number of 1024-word blocks. To release memory, a MME GEMREL is executed. When the MME GEMREL is successful, a type 011 record is written to report the number of words released. To release peripherals, a MME GERELS is executed. When the MME GERELS is successful, the same device information as reported in SCF record 015 is recorded in type 011.

During execution, a job may also dynamically create, delete, release, allocate, or modify catalogs or file structures through the use of FMS. The appropriate SCF records report these activities.

A job may perform input and output operations through either the Input/Output Supervisor (IOS) or GFRC. While using either, the occurrence of each I/O error results in an SCF record type 009 which contains the device description, the number of connects, the number of errors, the error ratio, hardware statuses, and Any IOM command. I/O error in reading or writing a permanent file results in an SCF record type 007, which is written at when the file is deaccessed. This record includes the SMC name, the filename, and file classification. Any security violation error detected by FMS results in an SCF record type 008 which contains the error code, device name, and file classification.

A job in execution may dynamically snapshot dump portions of its memory. The dump produced is normally written to the job's P* or a dedicated printer. If there is no P* or dedicated printer allocated, the dump is written to the SCF in multiple type 013 records. Each SCF record 013 contains one line of dump output formatted for printer output.

1.1.2.6 Termination

Upon activity termination, whether normal or abnormal, the execution report for the activity is generated, memory allocated to the activity is released, peripherals allocated to the activity are deallocated, and an SCF record type 015 is written. The type 015 record summarizes the activity statistics: type of termination, memory usage, time limit and count, SYSOUT information, and peripherals. Multiple type 015 records are written if one record is not adequate in size.

If the terminating activity is the last activity of the job, the job is released for system output. If not the last activity, the next activity is scheduled for peripheral allocation. If the termination is abnormal and due to console operator request, the job is terminated as if it were the end-of-job. If the termination is abnormal and is not due to console operator request and is not end-of-job, an abort indicator is set, and subsequent activity allocation will be scheduled only for language processing activities.

1.1.2.7 System Output

All output written on the System Output File is transcribed at the end of a job into hard copy for printer and/or card punch by SYSOUT. The output may be directed to the originating location of the job, to a user-specified remote device, or to the central site print/punch devices. When the output for a job is transferred to the output media, an SCF record type 014 is written. This record summarizes the description of the device to which the output is sent, the number of reports, and the number of records.

1.1.2.8 DRL TASK

DRL TASK was created to spawn a job to the FORTRAN compiler but may be used for other applications. DRL TASK first copies the PATs for the files in the file list to the user's buffer. The buffer contents are then written on the first two blocks of the user's *J file. The job then enters the batch environment at the memory allocation level, bypassing system input and system scheduler.

Just as the memory allocator is enabled, an SCF record type 017 is generated to record the GELOAD limit requested, processor limit requested, urgency, activity mnemonic, ident, and SCCs. Once the job is picked up by the memory allocator, it is treated as any other batch job and the previously described records are generated, if appropriate.

1.1.3 TERMINAL SESSION

1.1.3.1 Prototype WWMCCS Intercomputer Network (PWIN) Accounting

When a PWIN connection is opened, the user host Network Accounting Program (NAP) is called to collect information about the user process, save it on the Network Access File (NAF), and pass it to the user Network Control Program (NCP) for transmission to the server host. The server host NAP is then called to augment the user record with information about the server process, record the record on the server host NAF, and pass the server information to the server NCP for transmission back to the user host. The user host NAP then locates the NAF record for the connection and adds to it the server host information. When the connection is closed, the server host NAP writes the connection NAF record to the SCF. The server host NCP then transmits the record to the user host where it is written to the user host SCF.

The connection SCF record on the user host is record type 101. The connection SCF record on the server host is record type 102. These records include the following information: user process type, server process type, network usage time, time of connect, server CPU time, user USERID, user SNUMB, server SNUMB, server host ID, user host ID, process security level, number of message units exchanged, message unit size, user connection ID, and terminal ID.

1.1.3.2 Line Control System

Once a remote terminal user establishes a physical connect to the system, the Line Control System (LCS) writes an SCF record type 103 (7.1 only) to indicate the physical terminal address. Before the user is permitted access to the system, the following sequence of identification and validation must be successfully completed:

```
USERID$PASSWORD?
PRIORITY?
IDENT?
SECURITY?
PROCESS?
```

If the user is unsuccessful in entering the information, at disconnect time the LCS issues a type 109 record (7.1 only) which records the disconnect reason.

If a priority level other than ROUTINE is indicated, an SCF record type 105 (7.1 only) is generated. Once an IDENT has been successfully entered, an SCF record type 104 containing the $IDENT image is written. After successful identification, the user is free to choose a process such as

TSS, TPE, or a DAC program.

During a terminal session, if the user switches from one process to another, a record type 107 (7.1 only) is written to the SCF. This record includes identification for both the TO and FROM process; in the case of DAC programs, this includes program SNUMB, unique identifier, program number, activity number, and USERID.

When a TSS user logs off, the TSS Executive produces an SCF record type 108 SCF record containing the date and time of log-on and resource statistics as the number of characters transmitted, the number of disk I/O requests, the core seconds used, and the calculated cost. When a terminal user logs off from a process other than TSS, LCP produces an type 106 (701 only) SCF record which records the same information.

1.1.4 SECURITY

With respect to security, the primary objective with the design of release SCF is to provide more meaningful security checks with additional flexibility for both the user and the site security officer. Some of the features of the WW7.1 security design are:

- Security Level
- User Profile System
- Locking of USERIDs

Prior to release WW7.1 the user always operated at his highest security level and the system had no security level. With release WW7.1, both the individual user and the system must declare security levels. Each individual User Security Level (USL) must be within the System Security Level (SSL). Whenever a USL is changed, a record type 207 (7.1 only) is generated which records the new USL. Whenever the SSL is changed, a type 205 SCF record (7.1 only) is generated which records the new SSL.

The User Profile System (UPS) uses permission bits to replace the concept of PRIVITY. Rather than granting PRIVITY to a user and giving him access to many potentially dangerous operations, the site security officer can grant each user permission to selected operations. Consequently, each user is assigned a User Permission Level (UPL).

Similar to the SSL, the System Permission Level (SPL) is defined by the site security officer. The SPL specifies the operations which are valid on the system. Whenever a UPL is changed a type 208 SCF record (7.1 only) is generated, and whenever the SPL is changed a type 206 SCF record (7.1 only)

is generated. Both of these records contain the new permission level.

Combined Attribute Blocks (CAB) are created by intersecting all known information (IDENT, SCCs, UPS functions requested and granted) about a job or terminal session. Whenever a CAB is altered, an SCF record type 209 (7.1 only) is written to describe the new CAB.

Prior to release WW7.1, resources were locked out of the system when a user located at that resource caused a security breach. The real offender, the user, was not penalized, while the resource, such as a terminal, was rendered useless until the console operator unlocked that resource. In release WW7.1, a user causing a serious security breach will be locked out of the system by having his USERID locked. Resources will be locked out only when they cause repeated security breaches and the user involved cannot be determined (e.g., the user has been guessing at USERID or PASSWORD). Whenever a USERID is locked or unlocked, SCF record type 213 (7.1 only) identifies the USERID. Whenever a terminal is locked or unlocked, SCF record type 214 (7.1 only) identifies the terminal.

Security markings for a batch job or terminal session are associated with an SCC. Whenever the SCC for a job or terminal session is altered, record type 210 monitors this alteration. Security markings can be designated for a single report to override that designated for the job as a whole. Whenever this is changed the SCF record type 211 (7.1 only) records the event.

Within N-Level Priority Processing, a job's priority level can be altered by the operator verb NLPPR. When this is done, SCF record type 212 audits the change.

A user program in execution may execute a MME .EMM (enter master mode) if UPS permission has been granted. When the MME .EMM is used, an SCF record type 215 is written. This record contains the activity type and address (both relative and absolute) of the MME.

Whenever a security check fails within any of the security functions, a type 204 SCF record (7.1 only) records the identity of the program receiving the failure, the program detecting the failure, and a keyword to establish the type of check. If this failure results in a security breach, a type 203 SCF record (7.1 only) is then written from the Security Control System to identify the program causing the breach, the program detecting the breach, and the breach message.

SCF record type 216 accounts for FMS or TSS attempts to create, modify, delete SMCs or PMCs and contains the SMC name, FMS function code, and a diagnostic message if the request was not successful.

The TSS Master Subsystem in WW7.1 allows the WASSO and operations personnel to perform privileged functions from TSS. These functions include:

UPDA	Update SMC
SMCL	List SMC
MUPD	Multiple update of SMC resources
PRIO	Authorize access and priority for restricted subsystems
MESS	WASSO sends all-points-message
TALK	Allows WASSO/console operator conversation
STUS	TSS status report
MONI	Terminal monitor
WHOS	List all users on TSS

Usage of these functions is recorded on an SCF record type 217 which includes the function code, the terminal ids of the monitor and monitored terminals, the USERIDs of the monitor and monitored terminals, the SMCs of the monitor and monitored terminals, the SCC of the monitored terminal, and any all-points-messages or WASSO/console operator conversations.

At any time during TSS operation, a user file access request may be an attempted breach of security functions. A security violation is recorded in an SCF record type 218 (7.1 only) which contains the TSS subsystem name, abort/reason, user's security matrix, U-S-I classification, filename, line classification matrix, and terminal identfication.

Record types 201 and 202 are written to record Network Security Violators (NSV). If only one host is involved in the connection request, only one NSV record is generated at that host. This record includes the following information: the time of day, USERID, terminal id, SCC, SNUMB, host and program number. If two hosts are involved in the connection request, two NSV records are written at each host, each record containing information pertinent to one of the hosts. If three hosts are involved in the connection such that a source host precipitates a connection attempt between the second and third host and a NSV occurs at the third host, the two NSV records will be written at the second and third hosts. At the source host, an SCF record will be written to record the three hosts involved.

1.1.5 SYSTEM FILES

Release WW7.1 requires new system files. Three files, LINETABLE, CAVEATIMAGE, and USER PROFILE, are specified by the installation at system startup time. The USAF (UPS Storage and Access File) is GEMOREd at system startup by the Security Control Program (SCP). The file LINETABLE contains the configuration for all communication end points, whether the end point is a device or process. The file CAVEATIMAGE contains the disk-resident copy of all security markings defined for the system. The file USERPROFILE contains the User Profile System information. The USAF file contains the security information for the active users on the system. Because of the file contents, activities against these files must be audited.

Changes to the files are monitored by SCF record types 301, 302, 303, (all in 7.1 only) and 304 which identify the file changed, the block changed, and records the block image before and after the change.

Whenever one of the files is either BUSIED or UNBUSIED, the job generating this condition, the file name and the block number are identified by SCF record type 305 (7.1 only).

If a block in one of these three files receives an error, SCF record type 306 (7.1 only) is generated to record the filename, type of error, block identification, and block image.

If one of these files is recovered from the duplicate file, SCF record type 307 (7.1 only) identifies the file and block and records both the bad and good block image.

Whenever a system program requires access to one of these files and must have a Peripheral Attribute Table (PAT) built for its use, a type 308 SCF record (7.1 only) identifies the file and requested permissions.

1.1.6 SYSTEM INSTRUMENTATION

The DATANET 355 handles the communications capabilities for the Honeywell Series 6000. The DATANET 355 receives and processes information from a variety of remote terminals to the Honeywell Series 6000, and then transmits output and control information from the Honeywell Series 6000 to these terminals. As a stored-program (GRTS/355) processor, the DATANET 355 relieves some of the workload from the Honeywell Series 6000 processor to provide greater system throughput. Consequently, total system monitoring requires communications records. The first DATANET 355 SCF record is type 401 (7.1 only), which is sent at the time of each

terminal physical disconnect. Record type 401 records the log-on time, disconnect time, the number of messages and characters sent/received, error ratios, and any exception conditions.

Whenever a console operator request has been validated by .MPOP7 and passed to the appropriate module to be executed, a type 402 record is written to report the console identification and the operator verb and/or message.

Whenever the console operator requests a system snapshot (SNAP), the dump is written to the accounting tape in multiple type 404 records which are similar to the type 013 records. Each SCF record type 404 contains one line of output formatted for printer output.

Whenever .MPOPM starts a program (e.g., starting of TSS or TPE), an SCF record type 403 is written to the accounting tape. This SCF record describes the program started and identifies the module which started it.

The allocation of tapes and removable disk packs generates console messages requesting the mounts. These mount requests are recorded as SCF record type 405. (Although the majority of these mount requests are from the Peripheral Allocator, mount requests can be from FMS, GFRC, GENC, or GEMORE Processor.)

1.1.7 PROCESS INSTRUMENTATION

At System Startup, the $CONFIG section of the bootdeck will be recorded on type 501 SCF records. Once Startup is completed the actual physical configuration will be automatically recorded on type 502 SCF records. The type 501 record will be the configuration card image, and the type 502 record will be the TYPFG console message image.

Whenever GCOS is restarted, an SCF record type 503 is written to indicate the point of restart of the SCF.

Upon detection of an SCF record type 001 for activity 1 or an SCF record type 015 for end-of-job, a type 504 record is generated. The type 504 record contains the system counters for processor idle time, number of jobs moved and swapped, number of reports, etc.

At site-determined discrete intervals, an SCF record type 505 (7.1 only) is passed from GRTS/355 to record the number of buffer refusals and the DATANET 355 idle processor time.

During TSS operations, TSS may allocate or grow user temporary disk files. When this is done an SCF record type

506 records the terminal identification, the device identification for the file, and the maximum size for the file.

During TSS execution, a statistical summary of TSS operations is produced at site-specified time intervals in an SCF record type 507. This record includes TSS system information and accumulated summary statistics such as TSS startup time and date, current time and date, number of lines, users, breaks, disconnects, subsystem kills, alarms, and swapouts.

During TSS execution, a more extensive statistical summary is reported in the SCF record type 508. This record is produced at the termination of each TSS subsystem and includes User Status Table (UST) information concerning the subsystem performance and usage. The type 508 record includes information such as time spent swapping the subsystem, time the subsystem is in memory, time the subsystem is awaiting memory allocation, subsystem size, subsystem processor time, number of swapouts, swapins, DRLs, and buffer reads.

In order for a job's output to be transcribed into hard copy for printer and/or card punch by SYSOUT, the JLIST table which contains report code information such as file code, report type (PRINT, PUNCH, REMOTE), report destination, etc. is utilized. SCF record type 509 (7.1 only) is generated for each job and contains the JLIST image.

Transaction Processing System (TPS) controls the concurrent execution of application programs providing online, interactive, or batch capability. Transaction Processing Application Programs (TPAPs) are normal user application programs which are initiated from a terminal by the user and scheduled by TPS. An SCF record type 510 is generated in response to either the console operator or master terminal command of TP STAT. This record includes accumulated summary statistics such as the number of rejected terminals, the number of rejected transactions, threshold counts for various tables and buffers, and information on each TPAP such as count of spawns, transactions accepted, and transactions completed.

The Total Online Test System (TOLTS) is a test and diagnostic system for maintenance of hardware. The system is invoked by the console operator, either by response to peripheral error messages on the console or by an operator input message via the console REQUEST button. As TOLTS executes, it generates type 511 SCF records which contain the formatted console messages.

Host Software Instrumentation (HSI) records (type 515) are written to the SCF in order to evaluate the reliability, performance, and efficiency of the PWIN software. The data collected provides information as to the time spent in sections of code, the frequency of use of sections of code, time between significant events, etc. The specific information recorded includes numeric event type, the time of event, and additional event related data such as SNUMB, number of queues in use, queue type, subqueue type, error codes, buffer size, terminal ID, remote DAC name, or remote host number.

SCF record types 512, 513, and 514 (all available for 7.1 only) are for the Security Control System (SCS) and Line Control System (LCS) to record instrumentation data.

The Security Control system, available in WW7.1, replaces the Security Control module which is used in release 6.4. Record type 934 (6.4 only) is generated whenever the Security Control module adds or changes the user security matrix.

SCF RECORD CLASSES

The ten SCF classes are defined as follows:

Class	Record Type Range
Batch	001 - 035
Terminal Session	101 - 135
Security	201 - 235
System Files	301 - 335
System Instrumentation	401 - 435
Process Instrumentation	501 - 535
User	601 - 635
User	701 - 735
User	801 - 835
User	901 - 935

The first digit of the record type represents the class; the remaining two digits are the record number (01-35) within a class. Record type 934 has been temporarily allocated to release WW6.4. This record will be replaced with a similar record in the 100 series in release WW7.1.

SCF CLASS - Batch Record Types 001-035

17 Records currently defined

OLD TYPE	NEW TYPE	ORIGIN	CONTENT	RELEASE*
5	001	System Input	Job Statistics	6.4,7.1
-	002	System Input	Job Priority	7.1
(26)	003	Peripheral Allocation	Activity Resources	6.4,7.1
18	004	FMS	File Allocation	6.4,7.1
17	005	FMS	User Function Audit	6.4,7.1
15	006	FMS	File Accounting	6.4,7.1
22	007	FMS	Prmfl I/O Error Accounting	6.4,7.1
21	008	FMS	Error Detection	6.4,7.1
3	009	IOS	I/O Error	6.4,7.1
(27)	010	Dynamic Media Allocation	Resource Accounting	6.4,7.1
(28)	011	Dynamic Media Release	Resource Accounting	6.4,7.1
18	012	FMS	File Deallocate	6.4,7.1
6	013	Program Dump/ Snap	Memory Image	6.4,7.1
2	014	SYSOUT	Report Statistics	6.4,7.1
1/8	015	Termination	Activity Accounting	6.4,7.1
-	016	Termination	Job Delete	6.4,7.1
-	017	DRL TASK	Task Start	6.4,7.1

( ) Indicates new SCF records to be implemented by Task 626

All of the above release WW6.4 records will be transferred to release WW7.1.

* Record types produced in the indicated WWMCCS system releases.

SCF CLASS - Terminal Session Record Types 101-135

9 Records currently defined

OLD TYPE	NEW TYPE	ORIGIN	CONTENT	RELEASE
64	101	PWIN	Connect (User Host)	6.4,7.1
65	102	PWIN	Connect (Server Host)	6.4,7.1
-	103	LCS	Connect Terminal	7.1
25	104	Terminal Log-on/LCS	Log-on Identification	6.4,7.1
-	105	LCS	Terminal Session Priority	7.1
-	106	SCS	Accounting	7.1
-	107	LCS	Terminal Line Switch	7.1
7	108	TSS	Time Sharing Log-off	6.4,7.1
-	109	LCS	Disconnect Reason	7.1

Record type 104 will have a slightly different format between releases WW6.4 and WW7.1.

SCF CLASS - Security Record Types 201-235

18 Records currently defined

OLD TYPE	NEW TYPE	ORIGIN	CONTENT	RELEASE
76	201	PWIN	User Host Security Breach	6.4,7.1
77	202	PWIN	Server Host Security Breach	6.4,7.1
-	203	SCS	Security Breach Message Text	7.1
-	204	SCS	Security Check Failure	7.1
-	205	SCS	Change System Security Level	7.1
-	206	UPS	Change System Permission Level	7.1
-	207	SCS	Change User Security Level	7.1
-	208	UPS	Change User Permission Level	7.1
-	209	UPS	Change Combined Block	7.1
-	210	SCS	Change SCC for Security Marking	7.1
-	211	SCS	Change SCC for a Report	7.1
-	212	N-LEVEL	Change Priority Level	7.1
-	213	SCS	Lock and Unlock USERID	7.1
-	214	LCS	Lock and Unlock Terminal	7.1
(30)	215	Slave Requests Master Mode	MME .EMM Audit (Slave)	6.4,7.1
16	216	FMS	Create/Delete SMC/PMC	6.4,7.1
24	217	Master Sub-System	Audit Function Usage	6.4,7.1
23	218	TSS	Security Violation	6.4,7.1

SCF CLASS - System Files Record Types 301-335

8 Records currently defined

OLD TYPE	NEW TYPE	ORIGIN	CONTENT	RELEASE
-	301	LCS	Change to Line Table	7.1
-	302	SCS	Change to Caveat File	7.1
-	303	UPS	Change to User Profile	7.1
-	304	UPS	Change to USAF	7.1
-	305	LCS, SCS, or UPS	System File Reason Busy/Unbusy & Who Caused Condition	7.1
-	306	LCS, SCS, or UPS	System File Identification for Error	7.1
-	307	LCS, SCS, or UPS	System File Identification and Description of Duplicate for Recovery	7.1
-	308	LCS, SCS, or UPS	System File PAT Creation	7.1

SCF CLASS - System Instrumentation Record Types 401-435

6 Records currently defined

OLD TYPE	NEW TYPE	ORIGIN	CONTENT	RELEASE
-	401	DATANET 355	Line Statistics at Disconnect	7.1
12	402	POP Main	Console Operator Verb Audit	6.4,7.1
-	403	GPOP	Program Started and Identity of Starting Program	6.4,7.1
6	404	System Snapshot	Memory Image	6.4,7.1
-	405	Tape/Disk Mount	Mount Request Audit	6.4,7.1
34	406	Resource Collection	Program Data, Peripheral Usage, & System Software Data	7.1

SCF CLASS - Process Instrumentation Record Types 501-535

15 Records currently defined

OLD TYPE	NEW TYPE	ORIGIN	CONTENT	RELEASE
-	501	Startup	Configuration	6.4,7.1
-	502	Startup Complete	Physical Configuration	6.4,7.1
9	503	Restart	Restart Sync Record	6.4,7.1
-	504	SCF Control Module	System Counters	6.4,7.1
-	505	DATANET 355	DATANET 355 Statistics	7.1
(29)	506	TSS Media Allocation	Temporary File Allocation	6.4,7.1
14	507	TSS	Statistical Summary	6.4,7.1
19	508	TSS	Test & Measurements at Subsystem Termination	6.4,7.1
-	509	SYSOUT	Printer/Punch Usage	7.1
10	510	TPE	Statistical Summary	6.4,7.1
4	511	TOLTS	System Start	6.4,7.1
-	512	SCS	Instrumentation	7.1
-	513	LCS	Instrumentation	7.1
-	514	LCS	Error Detection	7.1
63	515	PWIN	Host Software Instrumentation	6.4,7.1

SCF CLASS - User Record Types 901-935

1 Record currently defined

OLD TYPE	NEW TYPE	ORIGIN	CONTENT	RELEASE
20	934		Security Matrix Addition/Change	6.4

1.1.8 SCF RECORDING TECHNIQUE

In previous GCOS releases, a process wishing to generate an SCF record called the SCF processing module after formatting the SCF record. The SCF processing module then determined if the subject SCF record was to be recorded. A negative recording test resulted in a return to the user. The result of the foregoing sequence was that considerable time was wasted in:

- formatting a possibly unneeded record,
- calling the SCF control module,
- testing functions to determine if the record is wanted,
- returning to the caller if test results for SCF recording is negative.

The release WW6.4 implementation of SCF recording will eliminate the above deficiencies for recording of conditional SCF records by observing the following sequence.

1.1.8.1 Process-Oriented Tests

Processes producing conditionally recorded SCF records will precede the SCF formatting sequence with a macro (.SCFCK) which will determine if SCF formatting and a subsequent call to the SCF control module is required. If it is determined that recording of a record type is not required, the record formatting sequence and call to SCF control module are bypassed.

In some instances, data collection for SCF reporting consists of coding that is dispersed throughout a module. In this case, the .SCFCK macro is inserted in the code sequence prior to the call to the SCF Control module.

1.1.9 SCF CONTROL FUNCTIONS

The SCF Processor (.MSCF) is the focal point for all SCF record collections. As each of the previously mentioned system functions produce SCF records, they call upon .MSCF to:

- complete the standard header and
- accomplish the writing as required.

After each write to the SCF, checks will be made to determine if the write was successful or if end-of-file was detected. In either case, appropriate action is taken. Upon completion of its functions, the SCF processor returns to the caller. In addition to SCF record processing, a number of related functions have also been incorporated into .MSCF. These functions are listed below and will be detailed in the following paragraphs.

- SCF Module Initialization
- MME GESCF Processor
- SCF Processing and Security Trace Queue (STQ) Manager
- Unique Identifier Manager

1.1.10 SCF MODULE INITIALIZATION

This module will be invoked by Startup and will establish the operational parameters required by .MSCF as defined in the Startup Deck. The types of information required by .MSCF for initialization are:

- The identification of SCF records to be recorded.
- The identification of SCF records whose recording disposition cannot be changed.
- The identification of operator verbs to be journalized.
- The identification of SCF records to be placed in the STQ.
- Size of the STQ buffer.

1.1.11 JOURNALIZATION OF OPERATOR VERBS

If no specifications are delivered to Startup, no operator verbs, other than AUDIT, will be journalized. The console operator may alter the recording disposition of any valid operator verb from the operator console by specifying which verbs are to be recorded. Altering the disposition of record type 402(12) to no recording state, if a disposition change is allowed, will suspend recording of verbs until recording status is reinstated.

1.1.12 STQ REQUIREMENTS

Record types to be entered into the STQ must be identified. The records specified will be placed in the STQ for inspection by interested individuals. The size of the queue must also be specified.

1.1.13 SCF PROCESSOR

1.1.13.1 SCF Processing and STQ Manager

This module is invoked by .CALL. The calling sequence furnishes the address of the SCF record to be journalized. Based on Startup parameters, the record is placed in the SCF buffer for writing to the SCF file and the STQ. This module contains the following processing elements:

- SCF Processing
- STQ Management
- Unique Identifier Queue Purge

2.1.13.2 SCF Processing

Utilizing the furnished address to obtain the record, a test is made to determine if the record will fit into the output buffer. If space permits, the record is placed in the output buffer. Otherwise, a buffer write occurs prior to placement of the record. Output error detection and end-of-file processing are accomplished via a courtesy call.

1.1.13.3 STQ Management

A circular queue of SCF records will be maintained for inspection by interested individuals. Entries will be made in this queue based on specifications presented by Startup. The module will make entries in the queue until it is full and then roll to the beginning of the queue.

1.1.13.4 Unique Identifier Queue Purge

This function is performed by the Peripheral Allocator if it is determined that a job is not a candidate for execution due to detected errors.

1.1.13.5 Unique Identifier Manager

This module is invoked by GEIN for each new job received. GEIN will request the current Unique Identifier and saves the value in block 0 of J* for future reference. When a job is considered as a candidate for the Peripheral Allocator by the Scheduler, a call is made to the SCF Control Module with the location of SCF control values for that job. These values will be placed in the program number position in the Unique Identifier Queue (UIQ) along with the current Unique Identifier and the User Host Identification. Similarly, when a program is started by GCOS, a call will be made to the SCF Control Module to obtain a Unique Identifier and store this and other control values in the Unique Identifier Queue.

The module updates the Unique Identifier each time one is requested.

The current value of the Unique Identifier will be preserved in the event of a system failure and will be restored along with other SCF control parameters when the system is rebooted.

SECTION 2

RECORD NUMBER

INTRODUCTION

This section contains a description of the SCF records general content the conditions under which each is generated, and a listing of the GCOS module(s) which create and write the record.

This section does NOT contain the record formats. Record formats may be found in:

Statistical Collection File (SCF)
Record Formats Document
November 25, 1977

SCF RECORD NUMBER 001

SYSTEM INPUT

This record is generated when a job is initially accepted by the system. Input to the system may be from:

1. Local card reader
2. IMCV tape
3. Shared Mass Storage

    - Operator Spawn
    - MME GENEWS
    - Transaction Processor Spawn
    - Time-Sharing Spawn
    - Remote Batch

The record contains input statistics of job start and stop time, IOC time for the input device, and SCAN information.

This record is created and written by GEIN.

SCF RECORD NUMBER 003

PERIPHERAL ALLOCATION

This record is created after the completion of peripheral allocation for an activity. It reports the following events:

- Allocation of removable media (tape and packs)
- Allocation of permanent and temporary disk space
- Allocation of dedicated unit record devices
- PRIVITY requests

This record is created and written by ALC1.

SCF RECORD NUMBER 004

FMS FILE ALLOCATION

This record is created each time a user requests allocation of a file. The record identifies the requestor ans attributes of the file requested.

This record is created and written by FS37.

SCF RECORD NUMBER 005

FMS USER FUNCTION AUDIT

This record is written to record the creation, modification, or deletion of SMC/PMC entries. The record includes:

1. SMC name
2. FMS Function Code
3. A diagnostic message (if the request was not implemented successfully)

This record is created and written by FS37.

SCF RECORD NUMBER 006

FMS ACCOUNTING

This record is written to account for:

1. Space utilization after TSS restart (FS30)
2. After SMC/PMC modifications such as:

   - File create
   - File delete
   - File-size increase

The record includes the:

- SMC entry name
- FMS function code
- Space and time product
- Total resources
- Resources used

The record is created and written by either FS30 or FS31.

SCF RECORD NUMBER 007

PERMANENT FILE I/O ERROR

This record is created each time IOS detects an error in reading and writing a permanent file.

This record is created and written by FS50.

SCF RECORD NUMBER 008

FMS ERROR DETECTION

This record is created each time FMS detects an error in the file system. The record contains error code, file classification and device information.

The record is created and written by FS50.

SCF RECORD NUMBER 009

I/O ERROR ACCOUNTING RECORD

An I/O error statistical record is written whenever an error occurs on a peripheral channel or device. Each record contains the following information:

1. Time of day and time since startup
2. Number of connects on the device
3. Number of errors on the device
4. Status word setting after error
5. Device identity and position at time of error

This record is created and written by IOS.

SCF RECORD NUMBER 010

DYNAMIC MEDIA ALLOCATION

This module performs Dynamic Media Allocation/Deallocation accounting record formation for the Statistical Collection File. Requests for media allocation/deallocation via MME GEMORE, MME GEMREL, MME GERELS, and TSSK are processed by this module.

A type 010 record is written to record dynamic media allocation. The format of the record depends on the media requested.

1. Memory REQUEST

   - Size of memory requested

2. Tape REQUEST

   - File Serial Number
   - Device information

3. Mass Storage REQUEST

   - Current size of file
   - Maximum size of file
   - Device information
   - FMS information

The GCOS module POR5, MORE, MOR1 and AC10, process the various requests for media allocation and call the module PAAF. PAAF formats the record and calls MSCF to write it. In the case of a request for more memory, two other modules POP3 and POP8, must be modified to call PAAF to write a 010 record.

SCF RECORD NUMBER 011

DYNAMIC RELEASE OF MEDIA

This module performs Dynamic Media Allocation/Deallocation accounting record formation for the Statistical Collection File. Requests for media allocation/deallocation via MME GEMORE, MME GEMREL, MME GERELS, and TSSK are processed by this module.

This record records the dynamic release of media. The format of the record depends on the media being released.

1. Memory

   - Number of words lower memory released
   - Number of words upper memory released

2. Tape

   - Device information
   - File serial number
   - Channel time
   - Density
   - Record/file position

3. Mass Storage

   - Device information
   - Channel time
   - File size
   - Maximum file size
   - FMS information

4. Removable Mass Storage

   - Device information
   - Channel time
   - File size
   - Maximum file size

SCF RECORD NUMBER 012

FMS FILE OR MEMORY DEALLOCATION

This record is created each time a user requests the deallocation of a file. The record contains the information necessary for file usage accounting.

This record is created and written by FS37.

SCF RECORD NUMBER 013

SYSTEM SNAPSHOT

This record is generated when the following conditions exist:

1. A snapshot dump is requested of an area of a program's core memory.

2. There is no system output file (this is the normal default destination for a snapshot dump).

3. There is no dedicated printer (this is the second choice destination for a snapshot dump).

This record, which contains the dump information, is created and written by SNP3.

SCF RECORD NUMBER 014

SYSOUT REPORT WRITER

A SYSOUT report writer record is written as the job output is transferred to the output media. A record is written for media summarizing:

1. Destination media
2. Number of reports
3. Number of records.

This record is created and written by GEOT.

SCF RECORD NUMBER 015

ACTIVITY ACCOUNTING

The activity accounting record is written by BRT6 whenever an activity terminates (either normally or abnormally). The activity accounting record summarizes the activity statistics and includes:

1. The execution sources used by the activity (core size, urgency, etc.)
2. The files that were accessed
3. The security classification code
4. The activity disposition code

This record is created and written by BRT6.

SCF RECORD NUMBER 016

JOB DELETE

This record is written whenever a job is deleted from the system due to an error detected by wither GEIN, ALC1, or RGIN. The record contains the following information:

1. Deletion code (if available)
2. Error message
3. Module deleting job (GEIN, ALC1, or RGIN)

SCF RECORD NUMBER 017

SYSTEM INPUT

This record is generated when a job is spawned via a DRL TASK. The record contains the input statistics of job origin, ident security classification code, activity mnemonic, and the information pertaining to input.

This record is created and written by TSSK.

SCF RECORD NUMBER 101

CONNECT (USER HOST)

This record is created and written by PWIN. Information includes the following:

1. User process type
2. Server process type
3. Network usage time
4. Time of connect
5. Server CPU time
6. USERID
7. SNUMB (user and server)
8. Host ID (user and server)
9. Security level
10. Message information
11. Terminal ID

SCF RECORD NUMBER 102

CONNECT (SERVER HOST)

This record is created and written by PWIN. Information includes the following:

1. User process type
2. Server process type
3. Network usage time
4. Time of connect
5. Server CPU time
6. USERID
7. SNUMB (user and server)
8. Host ID (user and server)
9. Security level
10. Message information
11. Terminal ID

SCF RECORD NUMBER 104

TERMINAL LOG-ON IDENTIFICATION

This record is generated each time a user logs on. It contains the security classification marking for output, and the IDENT information.

The record is created and written by LOGN.

SCF RECORD NUMBER 108

TIME SHARING USER

The time sharing user record is generated whenever a time sharing user logs off the terminal. The record contains statistical information regarding the users time sharing session.

This record is created and written by TSSI.

SCF RECORD NUMBER 201

USER HOST SECURITY BREACH

This record is created and written by PWIN. It records network security violations. It includes time, terminal ID, USERID, SCC, SNUMB, host and program number.

SCF RECORD NUMBER 202

SERVER HOST SECURITY BREACH

This record is created and written by PWIN. Information includes the following:

1. User process type
2. Server process type
3. Network usage time
4. Time of connect
5. Server CPU time
6. USERID
7. SNUMB (user and server)
8. Host ID (user and server)
9. Security level
10. Message information
11. Terminal ID

SCF RECORD NUMBER 215

MME .EMM AUDIT

This record is generated each time a non-system program requests entry into master mode. The record contains the information necessary to properly identify the requestor and the terminal from which the request was made.

This record is created and written by FALT.

SCF RECORD NUMBER 216

FMS MASTER FUNCTION AUDIT

This record is generated each time an FMS privileged directive is used for a file. It contains the function code, file classification and user.

SCF RECORD NUMBER 217

MASTER SYSTEM USAGE JOURNAL

This record is written each time one of the master functions is exercised. The master commands and functions are:

MESS	WASSO sends all points messages
MONI	Terminal monitor
MUPD	Multiple update of SMC resources
TALK	Terminal to console communication
PRIO	Authorize access and priority for restricted subsystems
SMCL	List SMC entries
STUS	TSS status report
UPDA	Update SMC entries
WHOS	List all users on TSS

The master system usage8record has the following information

- Terminal ID being monitored (if any)
- Function code
- SMC entry name (if any)
- USERID of user being monitored (if any)
- SCC and SMC of terminal being monitored
- Text of messages sent to users of the console

This record is created and written by the following modules:

- TUPD
- TSMC
- TMUP
- TPRI
- TMES
- TTAL
- TSTU
- TMON
- TWHO

SCF RECORD NUMBER 218

TSS SECURITY VIOLATION

This time sharing record is written each time TSS detects a security violation. It contains:

- TSS subsystem name
- Abort code
- User Security Matrix
- USI classification
- Catalog/file structure (if any)

This record may be created and written from either TSSK or LOGN.

SCF RECORD NUMBER 402

CONSOLE OPERATOR VERBS

This operator interface records is written each time that the console operator types in a valid command on the system console. Operator commands may or may not be recorded based on the verb bit map and the particular bit associated with a given verb.

The record contains the standard header, the verb name, and a maximum of two words of verb text.

Record type 402 has an alternate format for the audit function, SCFAUD, is a record containing the standard header, a dummy verb name, "AUDMSG", and up to a maximum of 72 characters of free text message typed in by the console operator.

SCF RECORD NUMBER 403

GPOP PROGRAM START

.MPOP5 is the module responsible for bringing a program intomemory. If the program to be started was not passed to it from the Peripheral Allocator (.MALC1), a system module number will be present in the start job request. If this module number is not zero and not .MGEIN, then .MPOP6 entry point 6 is called to set a unique job identifier and to write a 403 record.

SCF RECORD NUMBER 404

PROGRAM DUMP

This record is generated when the following conditions exist:

1. A "SNAP" is requested by the console operator.
2. There is no dedicated printer (this is the first choice destination for a "SNAP").

This record, which contains the dump/SNAP information is created and written by SNP3.

SCF RECORD NUMBER 405

TAPE/DISK MOUNT REQUEST AUDIT

The 405 record is written when a media mount is requested of the operator. Only certain mount requests are recorded for this record. There are media mount requests generated by PALC, GENC (IMCV tape mounts) and MOR1 (GEMORE magnetic tape). The record contains the following information:

1. File code
2. Reel number/pack number (if available)
3. Device information
4. Module requesting mount

SCF RECORD NUMBER 503

RESTART ACCOUNTING

This record is written whenever a system restart occurs and SCF continuation is requested.

SCF RECORD NUMBER 504

SYSTEM COUNTERS

The system counters record is built and written by .MSCF whenever a record type 1, 16, or 15 and last activity is encountered. This record preserves the content of several system counters at the time of job initiation/termination.

SCF RECORD NUMBER 506

TSS MEDIA ALLOCATION

This record is written in response to a time sharing request for temporary user file space.

This record is created and written by PAAF in response to a call from TSSK.

SCF RECORD NUMBER 507

TSS STATISTICAL SUMMARY

The TSS statistical summary record is periodically written (user option by setting .TSTAT). It contains summary statistics such as:

1. Number of users
2. Number of lines
3. Number of swapouts
4. Percentages of interactions by time
5. Percentage of interactions by program size
6. Percentage of interactions by file I/O

This record is created and written by TSRI.

SCF RECORD NUMBER 508

TSS TEST AND MEADUREMENT

This record is created each time that a time sharing subsystem is terminated. When termination occurs, selected information from the User Status table is written to the SCF type 508 record.

This record is created and written by TSSL.

SCF RECORD NUMBER 510

TRANSACTION PROCESSING SYSTEM STATISTICAL RECORD

The TPE statistical record provides accumulated usage statistics in response to either console operator or master terminal command of TPSTAF.

This record is created and written by TRXC.

SCF RECORD NUMBER 511

TOTAL ONLINE TEST SYSTEM

The records generated by the Total Online Test System reflect the text of the formatted console messages generated during its execution.

These records may be written by any of the following modules: TOLT, POLT, COLT, MOLT, or SOLT.

SCF RECORD NUMBER 515

CONNECT (SERVER HOST)

This record records information required to evaluate the reliability, performance, and efficiency of the PWIN software.

This record is created and written by PWIN.

11/21/77

SCF RECORD NUMBER 934

USER SECURITY MATRIX ADDITION/CHARGE

This record is written whenever:

1. An existing User Security Matrix is modified.

2. A new User Security Matrix is added.

The User Security Matrix record is created and written by FS50.

SECTION 3

GCOS MODULES

MODULE FUNCTION - Definition of GCOS III Macros

MODULE MODIFICATIONS

Addition of the .SCFCK macro for use by system programs generating SCF records. The macro will determine if the specified record type is to be recorded/traced unconditionally or if the specified record type is to be unconditionally recorded.

REMARKS

This macro has two formats:

FORMAT 1 - Conditional recording/trace of the specified record type.

ARG1 = SCF Record Type
ARG2 = Transfer Location to bypass SCF module call

A. SCF Record formatting consists of an inline code sequence.

```
        .SCFCK  ARG1,ARG2
           .
           .
           .    FORMATTING CODE SEQUENCE
           .
           .
        LDX1    (SCF Record Location)
        .CALL   .MSCF,2
ARG2    NULL
```

B. SCF Record formatting code is dispersed.

```
           .
           .
           .    CODE LEADING TO CALL SEQUENCE
           .    FOR .MSCF
           .
           .
        .SCFCK  ARG1,ARG2
        LDX1    (SCF RECORD LOCATION)
        .CALL   .MSCF,2
ARG2    NULL
```

FORMAT 2 - Unconditional recording of the specified record type.

ARG1 = SCF Record Type
ARG2 = Null
ARG3 = U

Format 2 of the .SCFCK should be coded in the initialization section of modules electing this option since the code must be executed only one time. The code sequence sets the recording status to on and the disposition to cannot change, for the specified record type. If this option is elected and tracing of the specified type is desired, initial specifications must be given at startup time or furnished at a later time via the operator's console.

FOR GCOS-RELATED FUNCTIONS

MODULE FUNCTION - GCOS III Module and Bit Definitions

MODULE MODIFICATIONS

A .MDSET was added to define the SCF control module .MSCF, the Peripheral Accounting module, .MPAAF and the SCF verb module, .MSCFV.

FOR GCOS-RELATED FUNCTIONS

MODULE FUNCTION - GCOS III Communication Region definition.

MODULE MODIFICATION

Define new communication region for .CRACF and expand it to 19 words.
Change .CF026 and rename old .CRACF region .CF027.

FOR RECORD TYPE 010

ASSOCIATED MODULE FUNCTION - Dynamic PRMFL Access/Deaccess

ASSOCIATED MODULE MODIFICATIONS

Obtain PRMFL PAT pointer and call .MPAAF.

FOR RECORD TYPE 011

ASSOCIATED MODULE FUNCTION - Dynamic PRMFL Access/Deaccess

ASSOCIATED MODULE MODIFICATIONS

Obtain PRMFL PAT pointer and call .MPAAF.

FOR GCOS-RELATED FUNCTIONS

MODULE FUNCTION

This module is invoked when an SCF tape exchange is required. The module has three entry points which function as follows:

EP1 - Request to exchange tape files from .MSCF courtesy call.

EP2 - Tape error, exchange tape files from .MSCF courtesy call.

EP3 - Operator request to exchange tape files from .MPOP9,2.

Entry Point 1 - Invoked by .MSCF courtesy call when an EOT is encountered on the current SCF file. An EOF followed by a trailer label will be written to the current file. If an alternate tape handler is assigned, it is requested. Beginning tape labels are written to the new tape file.

Entry Point 2 - Invoked by .MSCF courtesy call when an unrecoverable error is encountered. The operator is notified and an attempt is made to properly terminate the tape. A new tape is requested as in EP1 and, upon return to the courtesy call routine, the error record is written.

Entry Point 3 - Invoked by .MPOP9,2. This entry point processes the operator request in a manner similar to EP1, except that no action is taken until the file is not busy writing.

MODULE MODIFICATIONS

Alter references to .CRACF region where required due to reorganization of this area.

FORMATTING MODULE FUNCTION

The Peripheral Allocator schedules and allocates all peripherals used by slave programs or by an Input Media Conversion (IMCV) tape, a time sharing program, or by GEPR to exchange magnetic tape units.

When a new job is received the J* file, containing all control cards, is scanned to determine whether the job deck was made up correctly and which, if any, removable media should be retrieved. During this scan, the number of executions and number of $ BREAK cards are accumulated for later decision making. The worst case is determined for memory requirements, mass storage space and removable media needs, and process time and output line limits are accumulated. These values are then optionally used to set urgency and to gauge job length.

At the end of the initial scan of the job control file, the job is deleted if any errors were found, put in limbo if special resources need retrieval, or made a candidate for allocation if neither of these is true. If the job is lengthy and not urgent, it is bypassed (placed in Sieve status), and the operator is informed of this status. If a RUN request is not received within five minutes, the operator is reminded of the job status. The first time an activity is considered for allocation, the J* file is scanned from its current position to the next activity delimiter to accumulate exact memory and peripheral requirements. The file control card information is extracted and condensed into a peripheral requirement summary specifying each type of device. In addition, a peripheral detail entry is constructed for each file specified by a control card or implied because of the type of activity. The peripheral summary is used to perform a gross resource test before any detailed allocation is attempted. The detail entries are used to eliminate the need for a character scan of variable fields each time allocation is attempted. Keeping a queue of jobs waiting for resources and attempting allocation more than once for each activity prevents hardware delays and maintains a high level of throughput.

Actual allocation is accomplished by matching a program's peripheral requirements to the resources available until all needs are satisfied. During this process, a Peripheral Assignment Table (PAT) is generated.

The PAT allows .MIOS to associate a user-specified file code with an entry in the SCT and, at I/O time, with a specific piece of peripheral hardware.

When allocation is successfully completed, an SSA image is written to the first 640 words of the job data file (*J), and an entry is made in

the core allocation queue so the activity can be loaded and executed. At that time, the job status is set to "allocation complete" and is not considered for allocation again until the current activity has terminated.

When an activity requires a removable media, such as tape or disk, which is not on-line, .MALC1 formats an operator message requesting the mounting of the required media on a particular device. When all required media have been mounted and all other peripherals allocated, an entry is made in the Core Allocator's queue and the activity becomes a candidate for execution.

FOR RECORD TYPE 003

FORMATTING MODULE MODIFICATIONS

After all peripherals have been allocated to an activity, .MALC1 collects initial accounting data before passing the activity to the core allocator. .MALC1 scans through the PAT collecting data to be written to *J. During this scan, the SCF record generation subroutine (SCFGN) collects data for each active file code and formats it for the SCf record write subroutine (SCFWT). Just prior to passing the activity to GPOP for execution, SCFWT finishes off the standard header, calculates the record length, and calls .MSCF,2 to write the record.

FOR RECORD TYPE 016

FORMATTING MODULE FUNCTION

When a new job is received, the J* file, containing all control cards, is scanned to determine whether the job deck is made up correctly. At the end of this initial scan of the job control file, the job is deleted if any errors are detected. When the job becomes a candidate for allocation, peripherals are allocated on an activity-by-activity-basis. The first time an activity is considered for allocation, the J* file is scanned from its current position to the next activity delimiter to accumulate peripheral requirements. A summary of the peripheral requirements is generated during this scan. This peripheral summary is used to perform a gross resource test before any detailed allocation is attempted. In addition, a peripheral detail entry is constructed for each file specified explicitly or implicitly. this eliminates a character-by-character scan of the variable fields each time allocation is attempted. If errors are detected during this phase of allocation, the activity is deleted from further consideration.

When a fatal error is detected during the various phases of peripheral allocation, .MALC1 writes an error message to SYSOUT indicating why the job or activity was deleted from the system. Modifications have been made to .MALC1 to capture these error messages. Only the first

fatal error message is saved if multiple errors are detected. Before the job or activity is deleted from the system, a check is made to determine if the SCF record type 016 is to be written. If so, the subroutine BLD016 is called to format the record and fill in fields in the standard header which are pertinent to the deleted job. These fields are as follows:

Word(s)	Bit(s)	Contents
8	0-29	SNUMB of deleted job
	30-35	Not used
10-11	0-71	USERID of deleted job
12	0-17	Program number of deleted job
	18-35	Activity number of deleted job

FOR RECORD TYPE 405

FORMATTING MODULE MODIFICATIONS

When a media mount request has been made to the operator, an indicator is set on the Job Control Table entry for that job. As .MALC1 processes entries in FIFO, a check is made to see if all devices are ready. When a removable media has been mounted on a device, the information about the media is captured for the SCF405 record. The tape/disk label is read, if possible, to determine the true tape/disk number. The subroutine BCD405 is called to format the 405 record. Certain fields in the standard header are filled in. These fields are USERID, SNUMB, ACTIVITY NUMBER, and PROGRAM NUMBER. BCD405 also fills in a field in record with ".MALC1" to indicate which GCOS module called for the mount. If the SCF405 record is to be written, a call is made to .MSCF,2 to accomplish this.

FOR RECORD TYPE 015

FORMATTING MODULE FUNCTION

.MBRT6 is a privileged slave program called into the first 5K of slave memory* to do the necessary housekeeping associated with termination of an activity. .MBRT6 may be called from either .MBRT2 or .MBRT5 via .MCAL1,2. It purges any files so specified, formats the activity accounting record, gathers the accounting data for the execution report, compacts the Peripheral Assignment Table with saved files, initiates pertinent operator instructions where needed, unloads any tape reels to be dismounted, updates and writes out the SSA image blocks onto J* and informs the core allocator of activity completion. .MBRT6 also determines whether other activities are to follow and prepares the EOJ messages (entry in .CRJOB queue) to be typed by .MPOPM.

The following routines comprise the .MBRT6 module:

- . PURGE Purge Files
- . TERM Gather Accounting Data
- . REPT Produce Execution Report
- . NOACT Compact PAT & Build Release List
- . NOME Prepare Operator Messages
- . RELS Release Peripherals
- . MULL SYSOUT-Operator Interface

The first task performed by .MBRT6 is to gather accounting data for resources which have been used or allocated. To accomplish this, the SSA image, as it appears at activity allocation, is read from the first three records of the *J file, and a comparison is made with the current SSA data and the System Configuration Tables. A type 015 accounting record is written to the Statistical Collection File, and an accounting report is formatted for the execution report. When these functions have been performed, a call is paid to a user-supplied accounting module .MBRT7 (if such a module exists).

The Peripheral Assignment Table (PAT) is then scanned, and all files which are not to be saved for a subsequent activity are released and the PAT compacted. Operator action or information messages are produced where appropriate. The system output collector (.MSYOT) is then informed of end-of-activity or end-of-job. An entry is made in the POPM queue and .MBRT6 exits by executing a .CALL .MDISP,11. The .CRPOQ gate is shut while making the POPM entry.

Bits 0-4, 6-17 of .SWIT are reset after each activity. .MBRT2 sets bit 18 of .STATE word on to allow .MBRT6 to execute a MME .EMM.

*If the slave job being terminated is over 17K, .MBRT6 will release all but 9K.

FORMATTING MODULE MODIFICATIONS

Modifications to the .MBRT6 module consisted of: 1) inclusion of the standard header, 2) addition of the SCC, and 3) the logic necessary to produce continuation records.

In the standard header, only words 1 and 13 are posted in the .MBRT6 module (rest are posted in .MSCF). These two words have the following format:

Word	Bit(s)	Contents	Source
1	0-17	Record size (variable)	Set by .MBRT6
	18-20	Continuation bits 000 = No continuation rec. follows 010 = Initial rec. (Continuation Follows) 100 = Intermediate Continuation Record 110 = Final Continuation Record	Set by .MBRT6
	21-35	Record type = 015	Set by .MBRT6
13	0- 5	FNP number (Blank - not used)	
	6-17	Blank - not used	
	18-23	Terminal type	From *J (word 55)

TERMINAL TYPE CODES

01-02	Reserved for the system
03	Remote computer
04	Teleprinter
05	DATANET 760 VIP - Screen size = 4 lines x 46 characters
06	DATANET 760 VIP - Screen size = 8 lines x 46 characters
07	DATANET 760 VIP - Screen size = 16 lines x 46 characters
10	DATANET 760 VIP - Screen size = 26 lines x 46 characters
11	765/775 Series VIP - Screen size = 22 lines x 46 characters
12	785 Series VIP - Screen size = 22 lines x 92 characters
13	7700 Series VIP - Screen size = 12 lines x 80 characters
14	7700 Series VIP - Screen size = 22 lines x 46 characters
15	7700 Series VIP - Screen size = 24 lines x 80 characters
16	BTT 7340, BTT 7340E Banking Teller Terminal
17	Data Capture Cassette for Banking Teller Terminal
20	2741 teletypewriter
21-22	Reserved for the system
23	RLP300 Remote Line Printer
24	Reserved for the system
25	Mass Store Link
26-27	Reserved for the system
30	MRS200 Document Handler

	31	DRD200 Document Handler
	32	DRD236 or DHU1604/8/12/16 Document Handler
	33-37	Reserved for the system
	40-50	Reserved for T and D
	51-60	Reserved for GCOS
	61-77	Reserved for customer use

Word	Bit(s)	Contents
		NOTE The terminal type recorded in this record depends upon the type of job submitted. For example, if a job is submitted with JOUT disposition or a $DAC card from a teletype, the terminal type will be picked up as 03 (Remote Computer) since output is to be directed back to the device.
	24-35	Terminal ID From *J (word 55)

The Security Classification Code was added to word 14 of the record.

In the new continuation record (see word 1, bits 18-20 above), the number of SYSOUT report entries word (same as word 40 in the initial record), will be placed in word 14 with the applicable peripheral and report entries beginning in word 15. This differs from the old 1/8 record in that not all of the information is repeated in the continuation record, only that portion contained in the standard header. Since the information recorded in this accounting record is also utilized to produce the job execution report, significant changes were necessary to both the TERM (Gather Accounting Data) and REPT (Produce Execution Report) routines to produce the continuation records in the new format.

FOR RECORD TYPE 511

FORMATTING MODULE FUNCTION - Communications Online Testing

FORMATTING MODULE MODIFICATIONS - The standard header was added.

FOR GCOS-RELATED FUNCTIONS

MODULE FUNCTION

This module performs a system dump and writes restart information when a system failure occurs.

MODULE MODIFICATIONS

These modifications alter the length of .CRACF to include new elements and rearrangement of existing elements, and assign a system release identifier of WW6.4 to be placed on the dump tape headers.

FOR RECORD TYPE 215

FORMATTING MODULE FUNCTION

When a fault occurs, the secondary fault vector (FPRC of .MFALT) is entered from the primary fault vector. This routine finds and saves the fault type and transfers to FLT (.MFALT). This routine in turn transfers to the proper routine determined by the fault type. In the case of a MME fault, FLT executes a .GOTO (thru .CRCAL+2) to the appropriate module and its entry point to process the MME.

FEMM (EP10 of .MFALT) is entered when a program executes a MME .EMM to request master mode privileges. Slave programs that are allowed this MME are termed privileged slave. The following steps are taken within FEMM:

1. First, the routine:

   a. Tests .CRSNB to see if request is being made by a slave program,
   b. Tests the program's .STATE to see if the abort-in-control bit is not set, and
   c. Tests (thru the SCFCK Macro) to see if SCF Type 216 records are being recorded.

   If any of the above is not true, writing of an SCF record is bypassed and control passes to step 4.

2. If a slave program is requesting master mode privileges and the abort-in-control bit is not set and SCF Type 216 is being recorded, the routine shuts a local gate (FEMMGT) and proceeds to build a Type 216 record. The record consists of the SCF standard header (SCF0 to SCF13) and three additional words (SCF14 to SCF16) describing the type of activity, the absolute and relative address of the MME instruction, the MBA and BER (zero, if not extended memory), whether the program was allowed Master mode privileges, and if so, whether the program was already in master mode.

   NOTE: An "extended" absolute address is formed between bits 30-35 of SCF14 (the BER value) and bits 0-17 of SCF15 (the absolute address) when extended memory is configured.

3. After building the record, a call is made to .MSCF,2 to write the SCF record. Upon return, the FEMMGT gate is opened and the routine proceeds to process the master mode request.

4. To process the Enter Master Mode request, the routine first examines bit 18 of the .STATE word to determine if a MME .EMM is allowed. If bit 18 is off, the MME is not allowed. The abort code is set to 23 (octal) in the Q-register, and a .GOTO .MBRT1,3 is executed.

   If the MME is allowed (bit 18 of .STATE is on), the IC and I is recovered from .SSTAK, and the I portion is examined to determine if the program is in master mode. If bit 28 is on (program in master mode), the IC and I is put back into .SSTAK and a .EXIT is executed.

   If bit 28 of the IC and I is Off (program in slave mode), bit 28 is turned On and the LAL is added to the IC portion of the IC and I is put back in .SSTAK and a .EXIT is executed.

5. If master mode is allowed, return is made to the location following the MME .EMM. X5, X6, X7 are set for master mode and all other registers are restored to their previous state.

FORMATTING MODULE MODIFICATIONS

This record did not previously exist, thus the "modifications" in this case are insertions of entirely new code to audit the entry into master mode.

Only words 1 and 13 of the Standard header are set in this module (the remainder are set in .MSCF). Word 1 contains the record size (fixed, 17 words in length) while word 13 contains blanks (information not applicable to this record).

The following tests are made to see whether or not an SCF record type 216 will be written:

1. SCFCK Macro (test if type 216 being recorded).

2. If program number < 17 (i.e., if System Program number, do not record).

3. Test if last 6 bits (30-35) of SNUMB entry (in .CRSNB) = zero. (If zero, do not record. Method to exclude GEIN, which is assigned program numbers down from 77 octal depending upon the number of card readers configured. User programs have 17 octal in these six bits.)

4. Test if abort-in-control bit (bits of .STATE word) is set. If set on, do not record. Designed to exclude an entry from being made everytime an activity goes into termination and .MBRT6 (the activity termination overlay program), requests master mode.

5. Finally, a patch table is included to exclude certain programs by their SNUMB name. This table currently has room for 6 entries, minus the two SNUMBs which are currently not being recorded (i.e., VIDEO and HEALS).

FOR GCOS-RELATED FUNCTIONS

MODULE FUNCTION

This module provides symbol definitions for assembly of File System modules.

MODULE MODIFICATIONS

Expansion of standard header is required to include newly defined elements.

FOR RECORD TYPE 006

FORMATTING MODULE FUNCTION

Module .MFS30 accounts for permanent file space utilization by preparing Statistical Collection File records. It also initializes device information after a Time Sharing System boot.

FORMATTING MODULE MODIFICATIONS

Append Standard SCF Header to record.

FOR RECORD TYPE 006

FORMATTING MODULE FUNCTION

Module .MFS31 updates accounting charges for users permanent files.

FORMATTING MODULE MODIFICATIONS - Standardize record format.

FOR RECORD TYPE 004/012

FORMATTING MODULE FUNCTION

Module .MFS37 audits FMS functions and also builds a PAT for nonstructured file allocation.

FORMATTING MODULE MODIFICATIONS

Obtain file and security information about files from FMS catalogs and format required areas of the standard header.

FOR RECORD TYPE 005

FORMATTING MODULE FUNCTION

Module .MFS37 audits FMS functions and also builds a PAT for nonstructured file allocation.

FORMATTING MODULE MODIFICATIONS

Obtain file and security information about files from FMS catalogs and format required areas of the standard header.

FOR RECORD TYPE 216

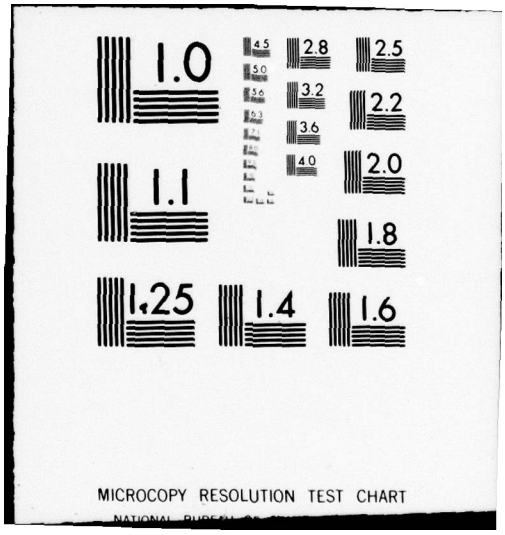
FORMATTING MODULE FUNCTION

Module .MFS37 audits FMS functions and also builds a PAT for nonstructured file allocation.

FORMATTING MODULE MODIFICATIONS - Standardize record format.

AD-A074 435 HONEYWELL INFORMATION SYSTEMS INC MCLEAN VA F/G 5/2
STATISTICAL COLLECTION FILE (SCF) MAINTENANCE MANUAL,(U)
NOV 77 D BEILFUSS, J BIELSKI, R DAY, R EWING DCA100-73-C-0055
UNCLASSIFIED NL

2 OF 3
ADA 074435


1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
4.5
5.0
5.6
MICROCOPY RESOLUTION TEST CHART

FOR RECORD TYPE 007

FORMATTING MODULE FUNCTION

Module .MFS50 performs all WWMCCS security accounting functions.

FORMATTING MODULE MODIFICATIONS

Append Standard SCF Header to record.

FOR RECORD TYPE 008

FORMATTING MODULE FUNCTION

Module .MFS50 performs all WWMCCS security accounting functions.

FORMATTING MODULE MODIFICATIONS

Include Standard SCF Header in record format.

FOR RECORD TYPE 934

FORMATTING MODULE FUNCTION

Module .MFS50 performs all WWMCCS security accounting functions and standardize record format.

FORMATTING MODULE MODIFICATIONS

Include Standard SCF Header in record format.

FOR RECORD TYPE 216

FORMATTING MODULE FUNCTION

Module .MFS77 performs System Master Catalog update and query functions.

FORMATTING MODULE MODIFICATIONS - Standardize record format.

FORMATTING MODULE FUNCTION

The GEIN Processor (.MGEIN) is the main GEIN module and controls local system input. Local system input is defined as that input originating from a card reader, Disk/drum, or IMCV magnetic tape. Input from the disk or drum may have previously originated from a remote device communicating via GERTS or NPS and the Remote GEIN Processor module (.MRGIN) or the time sharing system.

The GEIN Processor initiates and controls all input to the system from one of the three input media. It calls in one of the media modules as an overlay, and the combined modules start reading input data.

- Card Reader GEIN (.MGENA) - reads cards from the card reader into its input buffers. Two 270-word input buffers are contained within .MGENA. The 270 words consist of the contents of ten cards of data at a time, using the multi-record command. All input is in card image format.

- Disk/Drum GEIN (.MGENB) - reads input from the disk/drum into its two input buffers. The buffers within .MGENB are in standard system format (320-word blocks with logical records of the variable type). The first block contains the proper pointers and control words to the remaining data. Input contained on the disk/drum may have come from Remote GEIN and a remote terminal, or the time sharing system. Remote input of this type is considered remote batch, as opposed to the conversational type of input controlled by the GCOS-III time sharing system which is considered direct-access.

- IMCV Magnetic Tape GEIN (.MGENC) - reads from magnetic tape into its two input buffers in standard system format.

The GEIN Processor (.MGEIN) is enabled by GEPOP for any of the following reasons:

- A Special Interrupt is received from any nonallocated card reader.

- An entry is made in Core Allocators queue for jobs on mass storage (disk/drum or other shared device).

- Input is from an IMCV magnetic tape.

When enabled, .MGEIN is copied into core and given control. Then .MGEIN calls in .MGENA, .MGENB or .MGENC as an overlay to itself and

proceeds to initiate input from the allocated peripheral. The two modules (.MGEIN and one of the other three media modules) perform two main functions:

1. Separate the input into jobs (distinguished by a $ SNUMB card)

2. Create two files for each job:

   a. A file designated J* which contains the control cards describing the job and one-word pointer records pointing to subfiles on the data file. The control card file becomes the beginning of the execution report for the job.

      If errors are encountered, appropriate comment records will be entered in the control card subfile and no additional data subfiles will be constructed. When building of the job file is complete, it is given to RGIN to be passed on to SYSOUT for printing and will not become a candidate for execution.

   b. A file designated *J which contains the data subfiles for that job.

GEIN continues to build two files per job until it encounters an end-of-file condition. After each job is read in it is turned over to the System Scheduler.

The structure of GEIN is such that the main module (GEIN Processor, .MGEIN) contains the majority of routines that are necessary for system input. These routines can serve all media modules and therefore the code does not have to be repeated. Communication between .MGEIN and the media modules is through static transfer vectors located at the secondary SYMDEF entry, "COMMON". Within each media module are the subroutines to actually begin the reading of system input and control the two multiplexed input buffers. These routines begin at the same core locations for all three modules, but there are small differences in the routines based on the different input media.

When an end-of-file condition is encountered and all necessary housekeeping functions are complete, control is passed back to GEPOP and the core space occupied by GEIN becomes available to other programs.

Multiple copies of GEIN may be in execution at the same time, while the original resides within the GCOS-III file system, thereby increasing system input.

FOR RECORD TYPE 001

FORMATTING MODULE MODIFICATIONS

Much of GEIN's accounting data for record type 001 was reformatted, expanded and standardized. In addition, MSCAN processing information, spawn job origins, and remote job terminal IDs were provided, as well as more descriptive job origin data. Logic is also included to complete portions of the Standard Header for which information is not yet available to .MSCF.

Upon recognition of a new job, and the SNUMB record if found, .MGEIN will call .MSCF,3 to obtain a Unique Identifier for the job. The Unique Identifier and Network Information for PWIN jobs will be passed to other GCOS elements in block 0 of J*. The network information is in words 38-39. The unique identifier is in word 40.

FOR RECORD TYPE 016

During GEIN processing, incoming jobs may be deleted for a variety of reasons. If a job is deleted, a deletion message is written to SYSOUT indicating the cause of deletion. An error code has been assigned to the various deletion reasons. These codes and the associated error message are specified in the table below. The code is saved for both the type 001 record and the 016 record. In addition, the actual deletion message is saved for the 016 record.

After the 001 record is written, a check is made to determine if the type 016 is to be written. If it is, the same record buffer area is used. The deletion message is moved to the buffer area. The deletion code is also moved to the buffer area along with the module name ".MGEIN" to indicate which GCOS module deleted the job. The record header is filled in with information about the deleted job. A call to .MSCF,2 is made to write the record.

GEIN DELETION CODES

ERROR CODE	ERROR MESSAGE
THSR01	Illegal $ SNUMB, job deleted
THSR02	Bad $ IDENT card
THSR03	Valid $ USERID control card missing
THSR04	Multiple $ USERIDs are illegal
THSR05	Valid SCC missing on $ SNUMB
THSR06*	$ control card sequence error
THSR07	Card has illegal character
THSR08	Card encountered trouble
THSR09	Card has bad Checksum, operator aborted
THSR10	Card has bad Checksum, operator continued
THSR11	$ DKEND card missing
THSR12	Invalid $ SELECT syntax
THSR13	$ SELECT password incorrect
THSR14	Imbedded $ SNUMB in $ SELECT file illegal
THSR15	$ SELECT greater than 10 deep
THSR16	FMS error (several messages possible)
THSR17	Input nonsystem standard
THSR18	Unknown transliteration type
THSR19	*J pat tumbled - cannot continue
THSR20	IMCV read error
THSR21	Bad data format on IMCV
THSR22	IMCV read error, job deleted
THSR23	Not used
THSR24	Not used
THSR25	Fatal I/O error - job must be rerun

*This error could occur only on compressed deck (COMDK) or binary block (DECK).

FORMATTING MODULE FUNCTION

The GEIN Processor (.MGEIN) is the main GEIN module and controls local system input. Local system input is defined as that input originating from a card reader, Disk/drum, or IMCV magnetic tape. Input from the disk or drum may have previously originated from a remote device communicating via GERTS or NPS and the Remote GEIN Processor module (.MRGIN) or the time sharing system.

The GEIN Processor initiates and controls all input to the system from one of the three input media. It calls in one of the media modules as an overlay, and the combined modules start reading input data.

- Card Reader GEIN (.MGENA) - reads cards from the card reader into its input buffers. Two 270-word input buffers are contained within .MGENA. The 270 words consist of the contents of ten cards of data at a time, using the multi-record command. All input is in card image format.

- Disk/Drum GEIN (.MGENB) - reads input from the disk/drum into its two input buffers. The buffers within .MGENB are in standard system format (320-word blocks with logical records of the variable type). The first block contains the proper pointers and control words to the remaining data. Input contained on the disk/drum may have come from Remote GEIN and a remote terminal, or the time sharing system. Remote input of this type is considered remote batch, as opposed to the conversational type of input controlled by the GCOS-III time sharing system which is considered direct-access.

- IMCV Magnetic Tape GEIN (.MGENC) - reads from magnetic tape into its two input buffers in standard system format.

The GEIN Processor (.MGEIN) is enabled by GEPOP for any of the following reasons:

- A Special Interrupt is received from any non-allocated card reader.

- An entry is made in Core Allocators queue for jobs on mass storage (disk/drum or other shared device).

- Input is from an IMCV magnetic tape.

When enabled, .MGEIN is copied into core and given control. Then .MGEIN calls in .MGENA, .MGENB or .MGENC as an overlay to itself and

proceeds to initiate input from the allocated peripheral. The two modules (.MGEIN and one of the other three media modules) perform two main functions:

1. Separate the input into jobs (distinguished by a $ SNUMB card)

2. Create two files for each job:

   a. A file designated J* which contains the control cards describing the job and one-word pointer records pointing to subfiles on the data file. The control card file becomes the beginning of the execution report for the job.

      If errors are encountered, appropriate comment records will be entered in the control card subfile and no additional data subfiles will be constructed. When building of the job file is complete, it is given to RGIN to be passed on to SYSOUT for printing and will not become a candidate for execution.

   b. A file designated *J which contains the data subfiles for that job.

GEIN continues to build two files per job until it encounters an end-of-file condition. After each job is read in it is turned over to the System Scheduler.

The structure of GEIN is such that the main module (GEIN Processor, .MGEIN) contains the majority of routines that are necessary for system input. These routines can serve all media modules and therefore the code does not have to be repeated. Communication between .MGEIN and the media modules is through static transfer vectors located at the secondary SYMDEF entry, "COMMON". Within each media module are the subroutines to actually begin the reading of system input and control the two multiplexed input buffers. These routines begin at the same core locations for all three modules, but there are small differences in the routines based on the different input media.

When an end-of-file condition is encountered and all necessary housekeeping functions are complete, control is passed back to GEPOP and the core space occupied by GEIN becomes available to other programs.

Multiple copies of GEIN may be in execution at the same time, while the original resides within the GCOS-III file system, thereby increasing system input.

FOR RECORD TYPE 001

FORMATTING MODULE MODIFICATIONS

Much of GEIN's accounting data for record type 001 was reformatted, expanded and standardized. In addition, MSCAN processing information, spawn job origins, and remote job terminal I.D.s were provided, as well as more descriptive job origin data. Logic is also included to complete portions of the Standard Header for which information is not yet available to .MSCF.

Upon recognition of a new job, and the SNUMB record if found, .MGEIN will call .MSCF,3 to obtain a Unique Identifier for the job. The Unique Identifier and Network Information for PWIN jobs will be passed to other GCOS elements in block 0 of J*. The network information is in words 38-39. The unique identifier is in word 40.

FOR RECORD TYPE 016

During GEIN processing, incoming jobs may be deleted for a variety of reasons. If a job is deleted, a deletion message is written to SYSOUT indicating the cause of deletion. An error code has been assigned to the various deletion reasons. These codes and the associated error message are specified in the table below. The code is saved for both the type 001 record and the 016 record. In addition, the actual deletion message is saved for the 016 record.

After the 001 record is written, a check is made to determine if the type 016 is to be written. If it is, the same record buffer area is used. The deletion message is moved to the buffer area. The deletion code is also moved to the buffer area along with the module name ".MGEIN" to indicate which GCOS module deleted the job. The record header is filled in with information about the deleted job. A call to .MSCF,2 is made to write the record.

GEIN DELETION CODES

ERROR CODE	ERROR MESSAGE
THSR01	Illegal $ SNUMB, job deleted
THSR02	Bad $ IDENT card
THSR03	Valid $ USERID control card missing
THSR04	Multiple $ USERIDs are illegal
THSR05	Valid SCC missing on $ SNUMB
THSR06*	$ control card sequence error
THSR07	Card has illegal character
THSR08	Card encountered trouble
THSR09	Card has bad Checksum, operator aborted
THSR10	Card has bad Checksum, operator continued
THSR11	$ DKEND card missing
THSR12	Invalid $ SELECT syntax
THSR13	$ SELECT password incorrect
THSR14	Imbedded $ SNUMB in $ SELECT file illegal
THSR15	$ SELECT greater than 10 deep
THSR16	FMS error (several messages possible)
THSR17	Input nonsystem standard
THSR18	Unknown transliteration type
THSR19	*J pat tumbled - cannot continue
THSR20	IMCV read error
THSR21	Bad data format on IMCV
THSR22	IMCV read error, job deleted
THSR23	Not used
THSR24	Not used
THSR25	Fatal I/O error - job must be rerun

*This error could occur only on compressed deck (COMDK) or binary block (DECK).

FORMATTING MODULE FUNCTION

The GEIN Processor (.MGEIN) is the main GEIN module and controls local system input. Local system input is defined as that input originating from a card reader, Disk/drum, or IMCV magnetic tape. Input from the disk or drum may have previously originated from a remote device communicating via GERTS or NPS and the Remote GEIN Processor module (.MRGIN) or the time sharing system.

The GEIN Processor initiates and controls all input to the system from one of the three input media. It calls in one of the media modules as an overlay, and the combined modules start reading input data.

- Card Reader GEIN (.MGENA) - reads cards from the card reader into its input buffers. Two 270-word input buffers are contained within .MGENA. The 270 words consist of the contents of ten cards of data at a time, using the multi-record command. All input is in card image format.

- Disk/Drum GEIN (.MGENB) - reads input from the disk/drum into its two input buffers. The buffers within .MGENB are in standard system format (320-word blocks with logical records of the variable type). The first block contains the proper pointers and control words to the remaining data. Input contained on the disk/drum may have come from Remote GEIN and a remote terminal, or the time sharing system. Remote input of this type is considered remote batch, as opposed to the conversational type of input controlled by the GCOS-III time sharing system which is considered direct-access.

- IMCV Magnetic Tape GEIN (.MGENC) - reads from magnetic tape into its two input buffers in standard system format.

The GEIN Processor (.MGEIN) is enabled by GEPOP for any of the following reasons:

- A Special Interrupt is received from any nonallocated card reader.

- An entry is made in Core Allocators queue for jobs on mass storage (disk/drum or other shared device).

- Input is from an IMCV magnetic tape.

When enabled, .MGEIN is copied into core and given control. Then .MGEIN calls in .MGENA, .MGENB or .MGENC as an overlay to itself and

proceeds to initiate input from the allocated peripheral. The two modules (.MGEIN and one of the other three media modules) perform two main functions:

1. Separate the input into jobs (distinguished by a $ SNUMB card)

2. Create two files for each job:

   a. A file designated J* which contains the control cards describing the job and one-word pointer records pointing to subfiles on the data file. The control card file becomes the beginning of the execution report for the job.

      If errors are encountered, appropriate comment records will be entered in the control card subfile and no additional data subfiles will be constructed. When building of the job file is complete, it is given to RGIN to be passed on to SYSOUT for printing and will not become a candidate for execution.

   b. A file designated *J which contains the data subfiles for that job.

GEIN continues to build two files per job until it encounters an end-of-file condition. After each job is read in it is turned over to the System Scheduler.

The structure of GEIN is such that the main module (GEIN Processor, .MGEIN) contains the majority of routines that are necessary for system input. These routines can serve all media modules and therefore the code does not have to be repeated. Communication between .MGEIN and the media modules is through static transfer vectors located at the secondary SYMDEF entry, "COMMON". Within each media module are the subroutines to actually begin the reading of system input and control the two multiplexed input buffers. These routines begin at the same core locations for all three modules, but there are small differences in the routines based on the different input media.

When an end-of-file condition is encountered and all necessary housekeeping functions are complete, control is passed back to GEPOP and the core space occupied by GEIN becomes available to other programs.

Multiple copies of GEIN may be in execution at the same time, while the original resides within the GCOS-III file system, thereby increasing system input.

FOR RECORD TYPE 001

FORMATTING MODULE MODIFICATIONS

Much of GEIN's accounting data for record type 001 was reformatted, expanded and standardized. In addition, MSCAN processing information, spawn job origins, and remote job terminal IDs were provided, as well as more descriptive job origin data. Logic is also included to complete portions of the Standard Header for which information is not yet available to .MSCF.

Upon recognition of a new job, and the SNUMB record if found, .MGEIN will call .MSCF,3 to obtain a Unique Identifier for the job. The Unique Identifier and Network Information for PWIN jobs will be passed to other GCOS elements in block 0 of J*. The network information is in words 38-39. The unique identifier is in word 40.

FOR RECORD TYPE 016

During GEIN processing, incoming jobs may be deleted for a variety of reasons. If a job is deleted, a deletion message is written to SYSOUT indicating the cause of deletion. An error code has been assigned to the various deletion reasons. These codes and the associated error message are specified in the table below. The code is saved for both the type 001 record and the 016 record. In addition, the actual deletion message is saved for the 016 record.

After the 001 record is written, a check is made to determine if the type 016 is to be written. If it is, the same record buffer area is used. The deletion message is moved to the buffer area. The deletion code is also moved to the buffer area along with the module name ".MGEIN" to indicate which GCOS module deleted the job. The record header is filled in with information about the deleted job. A call to .MSCF,2 is made to write the record.

GEIN DELETION CODES

ERROR CODE	ERROR MESSAGE
THSR01	Illegal $ SNUMB, job deleted
THSR02	Bad $ IDENT card
THSR03	Valid $ USERID control card missing
THSR04	Multiple $ USERIDs are illegal
THSR05	Valid SCC missing on $ SNUMB
THSR06*	$ control card sequence error
THSR07	Card has illegal character
THSR08	Card encountered trouble
THSR09	Card has bad Checksum, operator aborted
THSR10	Card has bad Checksum, operator continued
THSR11	$ DKEND card missing
THSR12	Invalid $ SELECT syntax
THSR13	$ SELECT password incorrect
THSR14	Imbedded $ SNUMB in $ SELECT file illegal
THSR15	$ SELECT greater than 10 deep
THSR16	FMS error (several messages possible)
THSR17	Input *nonsystem* standard
THSR18	Unknown transliteration type
THSR19	*J pat tumbled - cannot continue
THSR20	IMCV read error
THSR21	Bad data format on IMCV
THSR22	IMCV read error, job deleted
THSR23	Not used
THSR24	Not used
THSR25	Fatal I/O error - job must be rerun

*This error could occur only on compressed deck (COMDK) or binary block (DECK).

FORMATTING MODULE FUNCTION

The operator initiates this process via a console command (IMCV or IMCV9). The module .MPOPB (221) processes this command and issues a message to the operator to mount the IMCV tape. It then enables .MGEIN to process the tape.

.MGENC is called by .MGEIN to actually read the tape. .MGENC reads the tape label (if present) and generates the SCF record type 405 if required.

FORMATTING MODULE MODIFICATIONS

FOR RECORD TYPE 405

After .MGENC attempts to read to IMCV tape label, a check is made to determine if the SCF record type 405 is written. If it is, the subroutine SCFBLD is called to format the record. The tape device physical address is determined by calling *GEFADD*. A call is made to .MSCF,2 to complete the standard header and write the record to the SCF.

FOR RECORD TYPE 014

FORMATTING MODULE FUNCTION

When .MGEOT receives a job, it notes which peripherals are required, and adds to the count of jobs waiting for those destinations (printer, punch, remote terminal). It then makes a pass through the control portion to see if any new jobs can be sent. Space is then allocated for a subprogram to process the output for the first job in the list waiting to be output.

After the subprogram is initialized, the control portion of .MGEOT initiates a read from the J* file for the job into a buffer of the subprogram and then looks for other jobs or subprograms to service. If it finds none, it roadblocks.

When the read is complete, a courtesy call is paid to the subprogram. The subprogram processes all the other reads/writes. If the control program is still roadblocked after the entire J* file has been processed by the subprogram, the .MGEOT program state is changed to relinquish control before ending the courtesy call.

When in execution, the .MGEOT control program looks at the state of all subprograms in turn. When all subprograms have been serviced, it will try to assign a device and subprogram for any waiting jobs. At the end of each job, a type 014 accounting record is written to the SCF. When there are no more reports or jobs waiting for a destination, the device is released and the state of the subprogram is set for not in execution.

When there are no subprograms in execution and no jobs waiting for devices, .MGEOT sets its urgency to zero and calls the Dispatcher. It is then a candidate for swapping, and is brought back and placed into execution when some later request enables it with its normal urgency.

FORMATTING MODULE MODIFICATIONS

Standardize record format by adding the Standard SCF Header and completing fields of the Standard Header for which information is not available to .MSCF.

GCOS-RELATED MODULE

MODULE FUNCTION

MGNEW processes a job spawn request. There are three entry points for MGNEW. EP1 is for slave mode entry (MME GENEWS). EP2 and EP3 are master mode entries. PWIN calls EP2 to spawn a job. A convention with PWIN has been established that the first two words of the required buffer will contain the PWIN User Host ID.

MODULE MODIFICATIONS

Due to size restrictions within the SSA module, certain functionalities were moved from MGNEW to MGNW2. Thses changes are discussed in the module modifications for MGNW2. Other changes were made so that if PWIN calls EP2, the User Host information will be extracted from the buffer and written on the $ SNUMB card for processing by GEIN. The format of the SNUMB record appears below:

1	8	16	61	72
$	SNUMB	SSSSS,un	Network information	

This module was extensively altered.

GCOS-RELATED MODULE

MODULE FUNCTION

MGNW2 completes the job spawn processing.

MODULE MODIFICATIONS

Due to size limitations in MGNEW, certain functionalities were moved to MGNW2. Two new entry points were added to MGNW2. EP1 (TMPFL) processes a spawn where the input source is a temporary file. EP2 (STREAM) processes an in-core request (i.e., the input is a BCI stream of control cards).

This module was extensively modified.

FOR GCOS-RELATED MODIFICATIONS

MODULE FUNCTION

This module processes the MME GEIDSE, the operator verbs IDSER and IDSEJ, and performs control functions for I-D-S user programs.

MODULE MODIFICATIONS

Deletion of MME GEIDSE, function 2, write to system journal file, and remove journal termination verbs IDSER and IDSEJ.

FOR RECORD TYPE 009

FORMATTING MODULE FUNCTION

FMTAR (Format Statistical Collection Record) is the routine within MIOS that is called by a channel module's interrupt handler to format a type 009 statistical collection record. The transfer to FMTAR is initiated by any status for which the channel module desires a type 009 record written. Among the statuses which do not cause a transfer are:

Exception processing reissuing an I/O
Test and Diagnostics entry
DATANET 355/6600 FNP exists on an IOM direct channel
Special interrupt exists
Special I/O command exists

Prior to calling the .MSCF module to write the SCF record, FMTAR assumes the identity of CALC (.MPOPM) by using its LAL and program number.

FORMATTING MODULE MODIFICATIONS

Modifications to this module include standardizing the record header.

Only Words 1 and 13 of the standard header are posted in this module (the remainder are posted by .MSCF). Record size is variable (may be either 27, 29, or 33 words in length depending on presence of extended status bytes and these lengths). Word 13 must be blank (information not applicable to this device).

FOR GCOS-RELATED FUNCTIONS

MODULE FUNCTION - Accounting File Recording Control, Entry point 13

MODULE MODIFICATIONS

This entry point is replaced by .MSCF,2. ACTFL and its initialization code, was deleted from .MMOS. In addition, entry point 14 is no longer used to hold pointers to SCF controls. Map pointers are now an integral part of .CRACF and the I/O queue entry skeleton is located via bits 0-17 of .CRACF+11.

FOR GCOS-RELATED FUNCTIONS

MODULE MODIFICATIONS

CONTROL CARDS

$ACCOUNT Control Card

All options will be recognized except that I-D-S journalization will not be allowed. Bit 17 of .CRACF will be set if I-D-S options are recognized to relay to .MIOS initialization that the I-D-S control routine .MIDSC must be loaded.

$ACCBUF Control Card

This control card will be replaced by the $SCFBUF control card.

$SCFBUF Control Card

This control card replaces the $ACCBUF control card.

$SCFBUF

This control card defines initial values for:

- SCF records to be recorded,
- SCF records whose recording disposition may be altered,
- SCF records to be placed in the SCF trace table,
- The SCF trace table size,
- SCF recording of operator verbs.

SCFBUF Control Card Format

```
$       SCFBUF  [RECORD/ARG1,.../,DISP/ARG1,.../]
                [,TRACE/ARG1,.../,TRACES/nnn/]
                [,VERBS/ARG2,.../]
```

OPTIONS

RECORD - ARG1 defines the initial SCF records to be recorded.

DISP - ARG1 establishes the initial SCF record recording dispositions. Records defined by ARG1 may have their dispositions altered during system execution.

TRACE - ARG1 specified which SCF records will be placed in the SCF Trace Queue.

TRACES - nnn is the size of the SCF Trace Queue. This value may not exceed 4095 words.

VERBS - ARG2 defines initial operator verb actions to be recorded on the SCF.

The default for any optional group is NONE (No Recording and All dispositions may be altered).

```
ARG1 = ┌   [ALL]
       |   [Record Type,...]
       |   [Cn[(Thru Record Type)],...]
       └
```

```
ARG2 = ┌   [ALL]
       |   [Cn[(Thru Record type)],...]
       |   [Verb number,...]
       |   [Verb Name,...]
       └
```

If ALL is specified, bits representing currently valid elements are set to 1 in the appropriate map.

RECORD TYPE - expressed as its 3-digit numeric equivalent.

Cn - Class number as a value from 0 to 9.

Thru Record Type - The maximum record type to be initially considered within a class. If not present, the highest legal value for the class is assumed. If the option Cn (Maximum Record Type) is exercised, bits for the specified class will be set thru the identified maximum type.

C1(15) would result in bits 1-15 for CLASS1 of the selected option to be set on. Remaining bits are set to zero.

In addition, a maximum record type and verb type by class table will be maintained in .MINIT. This table will be consulted during option map generation to ensure that illegal map bits are not set. Invalid specifications will be truncated for Cn(type) and ignored for Record Type, Verb Type and Verb Name. Appropriate error diagnostics will be generated.

Verb number - Verbs are identified in a manner similar to SCF records. The disposition of a class of verbs is controlled by referencing verb zero of a class. Verb number 200 would set the disposition bit on (bit 0 of word 2) for class 2 verbs. This bit turned on would allow the console operator to dynamically change the journalization status of all verbs in class 2 via use of the SCFVON and SCFVOF verbs.

The default selling for bit 0 of each word is off (not allowing the console operator this option).

Verb name - The GCOS acronym assigned to a verb.

.MINIT will resolve the specification for SCF control into a series of SCF specification maps which will become a part of .MSCF during its initialization. During translation, a table of greatest SCF/Verb number by class will be consulted to ensure that invalid bits are not set within the maps. There will also be a table of valid operator verbs ordered by class and corresponding mapping values for verb map resolution. The Trace Queue size will be translated into a binary value and saved, and .CRACF will be initialized. Reference to values resolved by .MINIT and required by .MSCF will be placed in .CRACF+2, bits 18-35. The module load table must be modified to include .MSCF nad .MPAAF.

The following chart depicts variables established by .MINIT for use by .MSCF.

GLOBAL VARIABLES REFERENCED BY .MSCF & ESTABLISHED BY .MINIT

VARIABLE NAME	INITIALIZED BY:	WHERE DEFINED	DESCRIPTION
ACNMSV	.MINIT	.MINIT	4 Words - Alphanumeric. If disk accounting is configured, contains the names of the two SCF files. File 1 name ACNMSC,+1 File 2 name ANCMSV+2,+3 If tape accounting is configured: Tape name ACNMSV+3 Characters 3-5
RMAP	.MINIT	.MINIT	10 Words - Binary Each word represents an SCF class. Each bit within a word represents the recording status of a record within a class. Bit 0 in all words must be zero. Bit(n) Value — Condition 0 — Do not record 1 — Record

```
                          .MINIT
                           RMAP
                    11111111|112222222222333333
bit - 012345678901234567|890123456789012345
      011111000000000100|000000000000000000
```

The above represents bit pattern of word 0 of RMAP if:

$ SCFBUF RECORD/1,2,3,4,5,15/

11/21/77

GLOBAL VARIABLES REFERENCED BY .MSCF & ESTABLISHED BY .MINIT

VARIABLE NAME	INITIALIZED BY:	WHERE DEFINED	DESCRIPTION
DMAP	.MINIT	.MINIT	10 Words - Binary Similar to RMAP except element controlled is SCF record recording mode changes. Bit(n) Value: 0 = Cannot alter initial mode; 1 = Initial mode may be altered
TMAP	.MINIT	.MINIT	10 Words - Binary Similar to RMAP except element controlled is SCF record trace. Bit(n) Value: 0 = Do not trace; 1 = Make trace entry
VMAP	.MINIT	.MINIT	10 Words - Binary Similar to RMAP except that the element controlled is recording of operator verbs. Bit(n) Value: 0 = Do not journalize; 1 = Journalize
TRSIZE	.MINIT	.MINIT	1 Words - Numeric (Binary) Size of SCF Trace Buffer

The above areas are defined contiguously within .MINIT using ACNMSV as a base reference.

GLOBAL VARIABLES REFERENCED BY .MSCF & ESTABLISHED BY .MINIT

VARIABLE NAME	INITIALIZED BY:	WHERE DEFINED	DESCRIPTION
.CRACF	.MINIT	GCOS Common	18 Words - Variable .CRACF defines the base word of the GCOS communication region in which SCF control elements are stored. Areas within .CRACF are addressed relative to this base. The following are established by .MINIT.

Word	Bit(s)	Contents	SCF File Device Disk	Tape
0	0	WRITING FLAG	0	0
	1	NOT INITIALIZED	1	1
	2	LOGICAL EOF ENCOUNTERED	0	0
	3	TAPE DEFINED	0	1
	4	DISK FILE 1 DEFINED	1	0
	5	DISK FILE 2 DEFINED	1	0
	6	CURRENT FILE IS TAPE	0	1
	7	CURRENT FILE IS DISK FILE 2	0	0
	8	CURRENT FILE IS DISK FILE 1	1	0
	9-16	ZEROS		
	17	= 1 if I-D-S CONFIGURED		
	18-35	ZEROS		
1				
2	0-17	PAT WORD 1 (SCT POINTER)		
	18-35	POINTER TO ASNMSV FOR USE BY .ISCF		
3	0-35	PAT WORD 2 (IOC TIME), SET TO ZERO		
4	0-35	PAT WORD 3 (CONTROL FLAGS), NOT ESTABLISHED		
5	0-35	PAT WORD 4 (POSITION)		
6	0-35	PAT WORD 5 (FILE DESCRIPTOR)		
7	0-17	ALTERNATE FILE PAT WORD 1		
8	0-35	ALTERNATE FILE PAT WORD 3, NOT ESTABLISHED		
9	0-35	ALTERNATE FILE PAT WORD 5		
10	0-35	ZEROS		
11	0-17	POINTER TO I/O ENTRY SKELETON (SET TO ZERO)		
	18-35	POINTER TO I/O QUEUE ENTRY		
12	0-35	UNIQUE ID (SET TO 1)		
13-17		SET TO ZEROS		

Example of $ SCFBUF and resolution Class 0 & 1 into maps:

```
$       SCFBUF  RECORD/001,002,015,C1(104),C2,
$       ETC     DISP/C0,C1/,
$       ETC     TRACE/C0,C1,C2/,TRACES/1280/,
$       ETC     VERBS/SCF,PATCH,C1(108),203/
```

CLASS 0 MAP

	11111111	1122222222223333
	01234567890123456	7890123456789012345
RMAP	0110000000000000	0000000000000000
DMAP	0111111111111110	0000000000000000
TMAP	0111111111111110	0000000000000000
VMAP	0110000000000000	0000000000000000

CLASS 1 MAP

	11111111	1122222222223333
	01234567890123456	7890123456789012345
RMAP+1	0111100000000000	0000000000000000
DMAP+1	0111111100000000	0000000000000000
TMAP+1	0111111100000000	0000000000000000
VMAP+1	0111111110000000	0000000000000000

To verbalize the above, SCF records initially recorded are, record types 001, 003 and 015 in SCF Class 0, Class 1 through type 104 and all of Class 2. Disposition changes are allowed for all records in Class 0 and 1. Recording disposition changes may not be made for Classes 2-9. All SCF records in Classes 0, 1, and 2 are traced. Operator verb action recording is specified for the verbs SCF and PATCH in Class 0, verbs in Class 1 thru 108 and Class 2 verb 203.

NOTE: Trace specifications are independent of recording specifications. Tracing will occur for any record that has the trace bit set to 1.

In addition, specification may be given as follows:

```
$       SCF     Option/Cl(009),015/
```

The above option specification may be interpreted as Class 1 thru record type 009 and record type 015.

FOR RECORD TYPE 104

FORMATTING MODULE FUNCTION

This module performs common log-on. It processes terminal log-on, assigns a Terminal Control Table (TCT), processes the USERID$PASSWORD, and checks security classification codes.

FORMATTING MODULE MODIFICATIONS

Include word 16 for terminal status and standardize record format. Include word 14 for the time that $ LOGN started processing this terminal.

FOR RECORD TYPE 218

FORMATTING MODULE MODIFICATIONS

The catalog/file string is provided in the accounting record for permanent file security violations. The remainder of the data was reformatted and the standard header was added.

FOR GCOS-RELATED FUNCTIONS

MODULE FUNCTION

.MMDMP is a slave program which translates the dump output of .MDUMP into edited print form onto a magnetic tape.

MODULE MODIFICATIONS

This program is generalized in nature and normally supports numerous GCOS releases. Macros were altered to eliminate support for all releases except WW6.4, G, and H. An identifier for this release was installed, WW6.4, and the definition of .CRACF was expanded to include newly defined elements. In addition, due to the implementation of .MMDMP macros, the SCF trace area cannot be separately defined. Therefore, .CRSTC was included in the definition of .CRACF.

A system release table was generated to include the new modules .MSCF and .MPAAF that were installed with this release. Release tables no longer required were deleted. TBLH will be used as the base to form TBLW.

FOR RECORD TYPE 511

FORMATTING MODULE FUNCTION - Memory Online Testing

FORMATTING MODULE MODIFICATIONS - The standard header was added.

MORE

FOR RECORD TYPE 010

ASSOCIATED MODULE FUNCTION - MME GEMORE Processor

ASSOCIATED MODULE MODIFICATIONS

Obtain file PAT pointer and call .MPAAF.

FORMATTING MODULE FUNCTION

.MMOR1 is an SSA module which processes the GEMORE request for magnetic tape. The requesting program passes the file code, reel number (optional), density code, and 7 or 9-track indicator. .MMOR1 builds the PAT body and PAT pointer if required. It requests a tape handler and determines if a mount message to the operator is required. If no message is required, control is returned to the requesting program. Otherwise, a mount message is built and issued to the operator's console. The module cycles through a loop until the device is ready. Control then passes back to the requesting program. .MMOR1 repeats the mount message if the operator does not respond after a specified time.

FORMATTING MODULE MODIFICATIONS

FOR RECORD TYPE 010

The PAT pointer for the requested tape is saved. A check is made to determine if record type 010 is to be written. If so, a call to .MPAAF,2 is made, passing the PAT pointer as an argument.

FOR RECORD TYPE 405

After a tape handler has been assigned and a tape mounted, if required, a check is made to determine if record type 405 is to be written. If it is, a call is made to .MLBL1,4 to read the tape label. The reel number is extracted for the SCF record if available. Otherwise, the tape number passed as an argument is used. The standard header is completed by .MSCF. A call is made to .MSCF,2 to write the record.

FOR RECORD TYPE 006, 010, AND 011

FORMATTING MODULE FUNCTION

This module performs Dynamic Media Allocation/Deallocation accounting record formation for the Statistical Collection File. Requests for media allocation/deallocation via MME GEMORE, MME GEMREL, MME GERELS, and TSSK are processed by this module.

FORMATTING MODULE MODIFICATIONS - N/A

FOR RECORD TYPE 511

FORMATTING MODULE FUNCTION - Peripheral Online Testing

FORMATTING MODULE MODIFICATIONS - The standard header was added.

FOR GCOS-RELATED MODIFICATIONS

MODULE MODIFICATIONS

The code to process the ACCEPT and IGNORE console verbs has been removed from this module. The ACCEPT function renamed SCFRON was moved to POQ8 and the IGNORE function renamed SCFROF was also moved to POQ8.

FOR RECORD TYPE 503

FORMATTING MODULE FUNCTION

Module .MPOPM contains the nonvariable portions of GPOP that always remain in core after system Startup (or restart) as privileged system Slave Program Number 1. Module .MPOPM is so constructed that system initialization, GPOP initialization, and system rollcall (including any restart procedure) are erased after Startup.

When GCOS has gained control, .MPOPM performs special initializing functions, the first of which is system rollcall. First, the status of each configured device is tested. If the status returned from the device is abnormal, the device is released from the system. This procedure brings all devices to a known status. When the status-checking procedures in rollcall are complete, special initialization procedures are performed. A check is made to see if device ST1 was defined and is available. If ST1 is not present, the system types a message and aborts. The initialization process continues by performing the following functions:

- Clears up the SMCs and builds available space tables for all shared devices,
- Sets up the .CRBUF table for mass storage allocation modules,
- Constructs an in-core llink table for each device,
- Processes the System Scheduler Clear response, and
- Initializes the SYSOUT/SYSOUT Report Writer tables.

System recovery of incomplete jobs and of SYSOUT tables then takes place. This is done only when the system is restarting. If the system is restarting, a type 503 record is written to the SCF. At system failure, certain tables were written to the SAVE file. These tables are read back into core and fall into these classes:

- SYSOUT tables
- Job tables
- Accounting file tables

From these tables, all uncompleted jobs are reentered into the system, and all SYSOUT activity is reentered. The accounting information buffers are replaced, and initialization of GCOS is complete.

FORMATTING MODULE MODIFICATIONS

Standardize record format. Allow for expansion of .CRACF.

FOR GCOS-RELATED FUNCTIONS

MODULE FUNCTION

The Hard Core Monitor, System Startup functions and the Operator Interface are included in .MPOPM. A number of related modules assist .MPOPM in performing these functions and are collectively referred to as GEPOP.

During the System Startup portion of GEPOP, initialization functions for the SCF are performed. Specifically, the SCF file is located if defined, an I/O Queue entry is obtained and the I/O skeleton in .MSCF is initialized. Tests are also made to determine if SCF continuations has been specified and appropriate action taken. Portions of .CRACF are initialized during execution of this code sequence.

MODULE MODIFICATIONS

Alters are required in Assign Accounting File to correct references to .CRACF. In addition, any IOTD references to .CRACF must be corrected to reflect the new length of the area (19 words).

GLOBAL VARIABLES REFERENCED BY .MSCF & ESTABLISHED BY .MPOPM

VARIABLE NAME	INITIALIZED BY:	WHERE DEFINED	DESCRIPTION
ACC.AB	.MPOPM	.MPOPM	640 WORDS - SCF I/O Buffer
.CRACF	.MPOPM	GCOS Common	19 WORDS - Variable POPM continues initialization of this area appending its values to those set by .MINIT and .MSCF.

Word	Bit(s)	Contents
0	0	WRITE IN PROCESS FLAG = 0
	1	NOT INITIALIZED FLAG (.MPOPM SETS TO 1)
	2	LOGICAL EOF FOUND = 0
	3-10	ESTABLISHED BY .MINIT
	11-15	ZERO (RESERVED)
	16	BTL WRITTEN (.MPOPM SETS TO 1)
	17	ZERO (RESERVED)
	18-35	CURRENT BLOCK SERIAL NUMBER
1	0-35	GATE
2	0-17	PAT WORD 1 (SCT) (.MINIT)
	18-35	POINTER TO ACNMSV IN .MINIT
3		PAT WORD 2 (IOC TIME)
4		PAT WORD 3 (CONTROL FLAGE) (.MPOPM)
5		PAT WORD 4 (POSITION) (.MPOPM)
6		PAT WORD 5 (FILE DESCRIPTOR) (.MPOPM)
7		ALTERNATE FILE PAT WORD 1 (.MINIT)
8		ALTERNATE FILE PAT WORD 3 (.MPOPM)
9		ALTERNATE FILE PAT WORD 5 (.MPOPM)
10	0-17	CURRENT BUFFER POINTER (.MPOPM)
	18-35	ALTERNATE BUFFER POINTER (.MPOPM)
11	0-17	POINTER TO I/O ENTRY SKELETON (.MSCF)
	18-35	CURRENT I/O ENTRY (.MPOPM)
12	0-35	UNIQUE ID (.MINIT, .MPOPM)
13-17		AS ESTABLISHED BY .MSCF

.CRETR	.MPOPM	GCOS Common	1 WORD - Binary Next SCF Seek Address

POPU

FOR GCOS-RELATED FUNCTIONS

MODULE FUNCTION

This module is invoked to complete the SCF tape exchange started in .MACTS. In addition, an entry point to communicate with the console operator is available. The two entry points defined for this module are as follows:

EP1 - Complete tape exchange

EP2 - Tape exchange inquiry

Entry Point 1 - Entry into this module is from .MACTS via a .GOTO. The module completes tape switching, writes the BTL and informs operations via a console message.

Entry Point 2 - Inquires of the operator if an exchange is desired because of an I/O error.

MODULE MODIFICATIONS

Alter references to .CRACF region where required due to reorganization of this area.

FOR GCOS-RELATED FUNCTIONS

MODULE FUNCTION

This module is invoked when an SCF disk file exchange is required. The module has three entry points:

EP1 - Request for file exchange from .MSCF courtesy call.

EP2 - Error detected, exchange files from .MSCF courtesy call.

EP3 - Operator request to exchange files from .MPOP9,2.

Entry Point 1 - Invoked by .MSCF courtesy call to exchange files when a logical EOF is encountered.

Entry Point 2 - Invoked by .MSCF courtesy call to exchange file when an unrecoverable error is detected. The operator is notified and an attempt is made to properly terminate the file. In the event that the file cannot be properly terminated, the number of valid records on the file is determined from the last good BSN and delivered to the purge program to control the dump to tape. Upon return to the courtesy call routine, the error record is written.

Entry Point 3 - Invoked by .MPOP9,2. This entry point processes the operator request to exchange files in a manner similar to EP1. File termination action is not taken until the file is not busy writing.

MODULE MODIFICATIONS

Alter references to .CRACF regions where required due to reorganization of this area.

FOR GCOS-RELATED FUNCTIONS

MODULE FUNCTION

This module is called via a .GOTO from .MPOPV. This module completes the file exchange and initiates the disk to tape purge program to dump the full file to disk. Appropriate messages to the console operator are also initiated.

MODULE MODIFICATIONS

Alter references to .CRACF region where required due to reorganization of this area.

FOR RECORD TYPE 403

ASSOCIATED MODULE FUNCTION

This module is responsible for starting job after core allocation has been completed. The module differentiates between system priviledged slaves and other slaves by testing for the presence of a module number. If a module number is present, the code sequence for loading priviledged system slaves is executed.

ASSOCIATED MODULE MODIFICATIONS

A call to .MPOP6,6 must be made after the program has been enabled by calling .MDISP,21. The return to .MPOP5 from .MPOP6 will be to the code sequence following the call for .MPOP6,6.

FOR RECORD TYPE 403

FORMATTING MODULE FUNCTION

A new entry point, EP6, will be defined for this module. Entry is via a .CALL from .MPOP5 after the system program to be logged has been enabled. Upon entry, the SCF control values will be obtained and the SCF record formatted and delivered to the SCF module for recording. A return is then made to .MPOP5.

FORMATTING MODULE MODIFICATIONS

Upon entry, a call will be made to .MSCF,3 to obtain a Unique Identifier for the program. A call will then be made on .MSCF,4 to table the Unique Identifier and SNUMB. The record area will be set to zeros. The standard header fields (record size, record type, SNUMB, and program number) of the job being started, are completed and the USERID field set to cord is then completed and a call is made on .MSCF,2 to record the record. A return is then made to .MPOP,5.

FOR RECORD TYPE 402

FORMATTING MODULE FUNCTION

Module .MPOP7 accounts for operator verb usage for the standard system verbs. SCF record 402 will be written from this module after determining that the request was valid and this particular verb is to be recorded.

FORMATTING MODULE MODIFICATIONS

Code was inserted to generate SCF record type 402 if the verb being processed is also being recorded. Code was inserted to perform the audit function, SCFAUD, and to format SCF record type 402 containing the audit message.

The system verb table has been removed from POP7 and installed in SRCH and the VOCAB verb has been removed from POP7 and repositioned in SRCH.

The IDSEJ and IDSER verbs have been deleted.

GCOS-RELATED MODULE

MODULE MODIFICATIONS

The code to process the ACSTAT and AUDIT console verbs has been removed from this module. The ACSTAT function was moved to POQ8 and the AUDIT function was moved to POP7. No other code in POP9 was changed.

FOR RECORD TYPE 011

ASSOCIATED MODULE FUNCTION

GEMREL services a MME GEMREL request to release memory for a slave program.

The amount of core to be release is compared to the amount the job has and at lease 1K must be left. If equal or more, the request is ignored. The core release may be either high, low, or slave service area. High release is accomplished by adjusting the slave limits word (.SALIM) and the memory use table, POMU. A test is made in high release to assure that the return address is not included in the block of core released. If this is the case, the request is ignored. Low core release makes necessary a move of the slave service area. All slave service area addresses that are relative to the lower address limit (LAL) must be made relative prior to moving and then made absolute after the move. The LAL table (.CRLAL) and memory use table (POMU) are adjusted to show the new LAL and the amount of additional available core respectively. Release of slave service area core is done if the amount requested for release does not include all of the slave service area. The memory usage table POMU is adjusted to show the released core available. The program is then enabled and control is returned to attempt further core allocation.

ASSOCIATED MODULE MODIFICATIONS

Obtain amount of core to release and call .MPAAF.

GCOS-RELATED MODULE

ASSOCIATED MODULE FUNCTION

This module implements the three renamed system verbs SCFROF, SCFRON, and SCFRST in addition to several verbs that already existed in the module.

FORMATTING MODULE MODIFICATIONS

The following is a description of the three SCF control verbs that have been installed in POQ8.

VERB - SCFROF

VERB FUNCTION - Allow changes in the recording status of an SCF record, record class or all SCF records, to do not record, as permitted by the disposition map.

VERB FORMAT

```
SCFROF (nnn   )
       {CLASSn}
       (ALL   )
```

Where:

nnn = a valid SCF record number
n = SCF record class (0-9)

VERB	ACTION REPLIES
nnn	INPUT ILLEGAL (INVALID RECORD TYPE)
nnn	DISPOSITION CAN'T BE CHANGED

If the verb argument is CLASSn or ALL, all SCF records whose disposition cannot be changed will be listed.

VERB - SCFRON

VERB FUNCTION - Same as SCFROF except that recording status is changed to record.

VERB - SCFRST

VERB FUNCTION - This verb is functionally the same as ACSTAT, which it replaces.

ASSOCIATED MODULE FUNCTION

GMORE (EP#1 of .MPOR5) services a MME GEMORE request for more memory for a slave program.

A check is made to determine if sufficient core is available. If sufficient core is not available, a job with zero urgency will be swapped out of core. If there is no job with a zero urgency, the request for core will be denied.

The program is tested for being in execution or in the Dispatcher queue. If it is in execution, .MPOP6, relinquishes until the job goes out of execution. It it is in the queue, it is removed. The core usage table POMU is scanned to see if there is a block of core just above the job large enough to satisfy the GEMORE request. If so, the additional core is allocated to the job immediately. The slave limits (.SALIM) words are changed, and the *POMU* table is adjusted to make that block of core unavailable. The core is cleared to zero and control returns to .MPOPM via a normal .EXIT.

If there is not sufficient space just above the job to satisfy the request, core compaction becomes necessary. The POMU table is scanned again to find the jobs that must be moved. If there is a core hole just below the job requesting core, it is necessary to move just one job (i.e., the one making the request). Otherwise, it may be necessary to move several jobs either upward or downward, depending upon conditions at the mement. Core compaction will not take place if program 3 (SYSOUT Report Writer) or program 5 (TSS) with a waiting courtesy call must be moved to satisfy the request.

When it is necessary to compact core, control returns to TPOP via .MPOPM to do the compaction before any further core allocation attempts are made. When compaction is finished, the GEMORE request will be reissued and allocated immediately, since the core compaction has made space available just above the job that made the request.

ASSOCIATED MODULE MODIFICATIONS

Obtain amount of additional core and call .MPAAF.

PROC/SOLT

FOR RECORD TYPE 511

FORMATTING MODULE FUNCTION - System Online Testing

FORMATTING MODULE MODIFICATIONS - The Standard Header was added.

QDIR

FOR GCOS-RELATED FUNCTIONS

MODULE FUNCTION - This module processes the following I-D-S calls:

.QBCD	Binary to Decimal
.QCLR	Clear Subroutine
.QCSM	Checksum Subroutine
.QDIR	Directive Processor
.QSFD	Scan Subroutine
.QMCH	Character Move
.QMEX	Write Message to Execution Report
.QWD	Move Words
.QPBK	I-D-S Journalization Subroutine

MODULE MODIFICATIONS

Code in .QPBK referencing SCF journalization was deleted.

QOPE

FOR GCOS-RELATED MODIFICATIONS

MODULE FUNCTION

Initializes the control block, the I-D-S control table, and determines if a JX file is defined for the caller. If a JX file is defined, it is opened for journalization activity. If JX is not defined and the SCF is assigned to tape, journalization activity is directed to the SCF.

MODULE MODIFICATIONS

If a JX file is not defined, .QOPE alters an NOP in .QDIR at .QPBK9 to a TRA to cause journalization to the SCF. I-D-S journalization to the SCF will not be permitted in WW6.4. Alter - to delete references to .QPBK9.

FOR GCOS-RELATED FUNCTIONS

MODULE FUNCTION

Defines values to be written by .MDUMP to save file.

MODULE MODIFICATIONS

The I/O control word is modified to include the new length of the SCF control region .CRACF.

RELS

FOR RECORD TYPE 011

ASSOCIATED MODULE FUNCTION - Peripheral Release Module

ASSOCIATED MODULE MODIFICATIONS - Obtain PAT pointer and call .MPAAF.

FOR RECORD TYPE 011

ASSOCIATED MODULE FUNCTION - Mass Storage Release

ASSOCIATED MODULE MODIFICATIONS - Obtain PAT pointer and call .MPAAF.

RGIN

FOR RECORD TYPE 016

FORMATTING MODULE FUNCTION

Jobs processed by .MGEIN are passed to .MRGIN for processing and scheduling according to the assigned class and priority.

If errors are discovered during .MRGIN processing, .MRGIN locates the SNUMB record and calls SYSOUT to produce an execution report for the DELETED SNUMB. Prior to calling SYSOUT, a job delete record is produced.

When the job is considered as a candidate for peripheral allocation, .MRGIN locates the SNUMB record for the job. Prior to presenting the job to the Peripheral Allocator, a call is made to .MSCF,4 with a pointer to the SNUMB record for the job. Entry into .MSCF will cause the SNUMB, Unique Identifier and Network information fields of the SNUMB record to be placed in S.TAB.

FORMATTING MODULE MODIFICATIONS

Alterations are required to ensure that when a RGIN error is detected, a type 016 record is formatted and presented to .MSCF for recording. If no RGIN errors are detected, a call is paid on .MSCF to table the jobs control information in S.TAB.

SCF

FOR GCOS-RELATED FUNCTIONS

MODULE FUNCTION

.MSCF is responsible for processing SCF recording and trace requests (from system modules), initiating SCF I/O, supplying Unique Job Identifiers to system modules and maintaining the Unique Job Identifier table. This module replaces .MIOS entry point 13 which previously supplied similar functions.

MODULE MODIFICATIONS

.MSCF is a new module. See Section 4 for a detailed module description.

FOR RECORD TYPE 504

MODULE FUNCTION

.MSCF module SYSCTR generates a type 504 system counters record when .MSCF is called to record/trace a type 1, 16, or 15 and last activity record.

FORMATTING MODULE MODIFICATIONS

.MSCF module SYSCTR is a new module. See section 4 for a detailed description.

FOR GCOS-RELATED FUNCTION

MODULE FUNCTION

This module implements the trace and verb console verbs.

Verb	Function
SCFTOF	TRACE SCF Record Off
SCFTON	TRACE SCF Record On
SCFVOF	Verb Recording Off
SCFVON	Verb Recording On
SCFVST	Verb Recording Status

SCF VERB FORMATS AND SYSTEM RESPONSES

VERB - SCFTOF

VERB FUNCTION - Turn off tracing of SCF records as specified by the argument.

VERB FORMAT - Same as SCFROF

VERB ACTION REPLIES - nnn INPUT ILLEGAL (ILLEGAL RECORD TYPE)

VERB - SCFTON

VERB FUNCTION - Same as SCFTOF except tracing is turned on.

VERB - SCFVOF

VERB FUNCTION - Allows changes in the recording status of an operator verb, verb or all verbs as permitted by the class disposition bit.

VERB FORMAT

```
SCFVOF (name  )
       {CLASSn}
       (ALL   )
Where:
  name = verb name
  n    = class (0-9)
```

VERB ACTION REPLIES

name DISPOSITION CAN'T BE CHANGED

(class disposition does not permit)
name ILLEGAL
CLASSn DISPOSITION CAN'T BE CHANGED

VERB - SCFVON

VERB FUNCTION - Similar to SCFVOF except that the SCF recording status is changed to on.

VERB - SCFVST

VERB FUNCTION - Reports verb recording status as requested by the argument.

VERB ACTION REPLIES
name ILLEGAL
name/status
Where:
Status = A currently recording on SCF
I currently ignored
C disposition can be changed
N disposition cannot be changed

Examples:
SCFVST LOCK/AC recording and disposition change allowed
SCFVST UNLOCK/IN not recording and disposition change not allowed
SCFVST GRANT/AN recording and disposition change not allowed

FOR RECORD TYPES 013 AND 404

FORMATTING MODULE FUNCTION

These record types are written to the SCF file when:

1. MME GESNAP is issued
2. There is no dedicated printer
3. There is no P*

In addition, the snapshot dump to the SCF may be requested by the console operator.

FORMATTING MODULE MODIFICATIONS

Standardize record format by adding the Standard SCF Header and add logic to determine which record type is being produced.

FOR GCOS-RELATED FUNCTION

MODULE FUNCTION

This module implements the search of the consolidated System and User Verb table. It is called by POP7 to indicate an operator command and it is called by SCFV to provide status and associated validation for SCF Verb Recording verbs.

FORMATTING MODULE MODIFICATIONS

Code was inserted to generate SCF record type 402 if the verb being processed is also being recorded. The list of valid verbs was regenerated in alphabetic sequence within class. A modified version of the Verb Validation Table generation macro follows:

```
Macro Name - VRB
Argument List
  ARG1 = Verb name
  ARG2 = Module name excluding .M
  ARG3 = Module entry point in ARG2 to process ARG1
  ARG4 = Verb Class (0-9)
  ARG5 = Verb number within Class (1-n) where n < 36 and > 0

Format
  VRB  ARG1, ARG2, ARG3, ARG4, ARG5
```

All of the above arguments are required.

```
Example:
  VRB  SCFCLO, POP9,2,0,2
```

FOR RECORD TYPE 511

FORMATTING MODULE FUNCTION - Total Online Testing Program

FORMATTING MODULE MODIFICATIONS - The standard header was added.

NOTE:

No changes were made to this module. It was reassembled with the alters for the Phoenix and CCTC patches because all of TRAX had to be reassembled to test TRXC.

NOTE:

No changes were made to this module. It was reassembled with the alters for the Phoenix and CCTC patches because all of TRAX had to be reassembled to test TRXC.

FOR RECORD TYPE 510

FORMATTING MODULE FUNCTION

Module TRXC provides the console/master functions for the Transaction Processing Executive.

FORMATTING MODULE MODIFICATIONS

The accounting data was reformatted and the standard header was added.

NOTE:

No changes were made to this module. It was reassembled with the alters for the Phoenix and CCTC patches because all of TRAX had to be reassembled to test TRXC.

NOTE:

No changes were made to this module. It was reassembled with the alters for the Phoenix and CCTC patches because all of TRAX had to be reassembled to test TRXC.

NOTE:

No changes were made to this module. It was reassembled with the alters for the Phoenix and CCTC patches because all of TRAX had to be reassembled to test TRXC.

NOTE:

No changes were made to this module. It was reassembled with the alters for the Phoenix and CCTC patches because all of TRAX had to be reassembled to test TRXC.

FOR RECORD TYPE 507

FORMATTING MODULE FUNCTION

TSRI is optionally used by the time sharing executive to write time sharing statistics to the Statistical Collection File.

FORMATTING MODULE MODIFICATIONS

The accounting data was reformatted and the standard header was added. The frequency with which TSRI generates the 507 record is set by the GCOS site at startup. Time intervals are specified in .TSTAT.

FOR RECORD TYPE 217

FORMATTING MODULE FUNCTION

Module TTAL allows the Master User to initiate messages to the Series 6000 console operator and receive responses if required.

FORMATTING MODULE MODIFICATIONS

The first message sent to the console by TTAL is stored in BCD in words 21 through n. The rest of the data in the record was reformatted, and the standard header was added.

FOR RECORD TYPE 217

FORMATTING MODULE FUNCTION

Module TMES allows the Master User to issue a message to all currently connected and subsequently connected terminals.

FORMATTING MODULE MODIFICATIONS

The message sent by TMES is stored in BCD in words 21 through n. The other data in the record was reformatted, and the standard header was added.

FOR RECORD TYPE 217

FORMATTING MODULE FUNCTION

Module TSTU allows the Master User to obtain a status summary of time sharing.

FORMATTING MODULE MODIFICATIONS

Record type 217 was reformatted and the standard header was added.

FOR RECORD TYPE 217

FORMATTING MODULE FUNCTION

Module TMON allows the Master User to monitor all traffic to and from a specified terminal.

FORMATTING MODULE MODIFICATIONS

Record type 217 was reformatted and the standard header was added.

FOR RECORD TYPE 217

FORMATTING MODULE FUNCTION

Module TWHO allows the Master User to obtain a listing of all users currently logged on to time sharing and the four-octal-digit line number to which they are connected.

FORMATTING MODULE MODIFICATIONS

Record type 217 was reformatted and the standard header was added.

FOR RECORD TYPE 217

FORMATTING MODULE FUNCTION

Record type 217 enables the Master User to add, delete, or update a System Master Catalog (SMC) entry.

FORMATTING MODULE MODIFICATIONS

Module TUPD was reformatted and the standard header was added.

FOR RECORD TYPE 217

FORMATTING MODULE FUNCTION

Module TSMC enables the Master User to list System Master Catalog (SMC) entries.

FORMATTING MODULE MODIFICATIONS

Record type 217 was reformatted and the standard header was added.

FOR RECORD TYPE 217

FORMATTING MODULE FUNCTION

Module TPRI allows the Master User to specify which users may access the CARDIN, LODX, LODS, and TALK subsystems and to set a priority level for CARDIN jobs.

FORMATTING MODULE MODIFICATIONS

Record type 217 was reformatted, and the standard header was added.

FOR RECORD TYPE 217

FORMATTING MODULE FUNCTION

Module TMUP enables the Master User to modify System Master Catalog (SMC) entries for maximum and used-to-date resources.

FORMATTING MODULE MODIFICATIONS

Record type 217 was reformatted and the standard header was added.

NOTE:

Module TSSA was updated with Level 14 patches. There were no SCF changes required for this module.

NOTE:

Module TSSA was updated with Level 14 patches. There were no SCF changes required for this module.

NOTE:

Module TSSA was updated with Level 14 patches. There were no SCF changes required for this module.

NOTE:

Module TSSA was updated with Level 14 patches. There were no SCF changes required for this module.

NOTE:

Module TSSA was updated with Level 14 patches. There were no SCF changes required for this module.

NOTE:

Module TSSA was updated with Level 14 patches. There were no SCF changes required for this module.

NOTE:

Module TSSA was updated with Level 14 patches. There were no SCF changes required for this module.

NOTE:

Module TSSA was updated with Level 14 patches. There were no SCF changes required for this module.

FOR RECORD TYPE 108

FORMATTING MODULE FUNCTION

TSSI performs utility functions for time sharing log-on, log-off, and command processing.

FORMATTING MODULE MODIFICATIONS

The accounting data was reformatted and the standard header was added.

NOTE:

Module TSSA was updated with Level 14 patches. There were no SCF changes required for this module.

FOR RECORD TYPE 17

FORMATTING MODULE FUNCTION

This portion of TSSK spawns a job into the batch environment, bypassing GEIN and PALC, and enters indirectly into the core allocator's queue.

FORMATTING MODULE MODIFICATIONS

A new record is written for all DRL TASK processing which includes date regarding the job's execution parameters, program number, and origin.

FOR RECORD TYPE 218

FORMATTING MODULE FUNCTION

TSSK is the Time Sharing Derail processor.

FORMATTING MODULE MODIFICATIONS

The catalog/file string is provided in the accounting record for permanent file security violations. The remainder of the data was reformatted and the standard header was added.

FOR RECORD TYPE 506

FORMATTING MODULE FUNCTION

This module performs Dynamic Media Allocation/Deallocation accounting record formation for the Statistical Collection File. Requests for media allocation/deallocation via MME GEMORE, MME GEMREL, MME GEMRELS, and TSSK are processed by this module.

FORMATTING MODULE MODIFICATIONS

Obtain PAT pointer and call .MPAAF.

FOR GCOS-RELATED MODIFICATIONS

MODULE FUNCTION

Allows time sharing to write a record to the Statistical Collection File.

MODULE MODIFICATIONS

Redundant information formerly provided by T.STAT was eliminated. Coding was added to provide the FNP (datanet) number for all TSS records written by DRL T.STAT.

FOR RECORD TYPE 508

FORMATTING MODULE FUNCTION

TSSL provides services for the various time sharing subsystems. Each time a user requests a specific subsystem, TSSL updates the user's UST (User Status Table), initiates the processing and manages any swaps and allocations of memory which may be required. Upon a user's termination in a subsystem, TSSL cleans up the UST, terminates the subsystem, and writes a type 508 SCF record concerning subsystem performance and usage.

FORMATTING MODULE MODIFICATIONS

The accounting data was reformatted and the standard header was added.

NOTE:

Module TSSA was updated with Level 14 patches. There were no SCF changes required for this module.

NOTE:

Module TSSA was updated with Level 14 patches. There were no SCF changes required for this module.

NOTE:

Module TSSA was updated with Level 14 patches. There were no SCF changes required for this module.

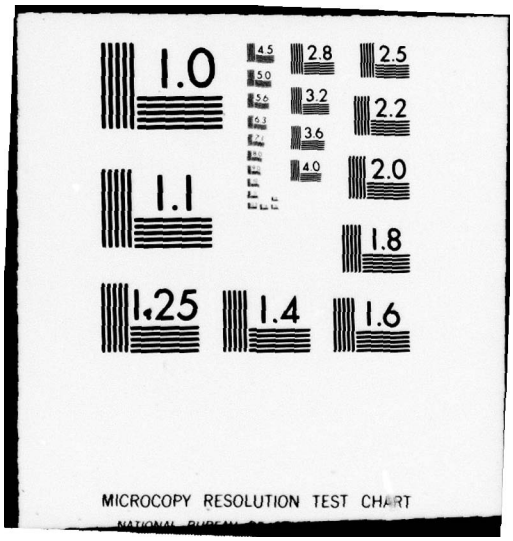
SECTION 4

MSCF HIERARCHICAL FUNCTION DIAGRAM

AD-A074 435 HONEYWELL INFORMATION SYSTEMS INC MCLEAN VA F/G 5/2
STATISTICAL COLLECTION FILE (SCF) MAINTENANCE MANUAL,(U)
NOV 77 D BEILFUSS, J BIELSKI, R DAY, R EWING DCA100-73-C-0055
UNCLASSIFIED NL

3 OF 3
ADA
074435

END
DATE FILMED
10-79
DDC


1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
4.5
5.0
5.6
6.3
MICROCOPY RESOLUTION TEST CHART

.MSCF
HIERARCHICAL FUNCTION DIAGRAM

	MODULE NUMBER	MODULE NAME
SCF CONTROL MODULE	776	.MSCF
MODULE INITIALIZATION	1.1	.ISCF
MASTER MODE ENTRY (MME GESCF) PROCESSOR	1.2	MMESCF
SCF RECORD & TRACE PROCESSING	1.3	CALSCF
UNIQUE JOB IDENTIFIER MANAGER	1.4	UJISCF
SCF I/O COURTESY CALL	1.5	CCSCF

MODULE INITIALIZATION 1.1 .ISCF

<This section is executed during startup
and then overlayed by the next module.>
Establish .CRSTC open gate word.
Initialize Unique Job Identifier
in .CRACF+12 to one.
Establish address of SCF Recording map
in bits 0-17 of .CRACF+13.
Establish address of SCF disposition
map in bits 0-17 of .CRACF+14.
Establish address of Operator Verb
recording map in bits 0-17 of .CRACF+15.
Establish address of SCF Trace map in
bits 0-17 of .CRACF+16.
Establish address of Sum map (recording
map or'ed with trace map) in bits
0-17 of .CRACF+18.
Establish address of map source in .MINIT
(.CRACF+2 bits 18-35).
Move map values from .MINIT to .MSCF.
Put absolute address of I/O entry
skeleton in .CRACF+11, bits 0-17.
Put SCF SCT address from bits 0-17 of
.CRACF+2 into bits 0-17 of skeleton
word 0.
Set absolute CC address, bit 24, and do
not abort, bit 28, flags in skeleton
word 0.
Put .MPOPM program number in bits 18-35
of skeleton word 1.
If DISK SCF
Perform DISKIO 1.1.1, else
Perform TAPEIO 1.1.2.
Set Master limits flag in bit 35 of
skeleton word 4.

	MODULE NUMBER	MODULE NAME
Set absolute address of courtesy call in bits 18-35 of skeleton word 7. Set absolute address of I/O status return words in skeleton word 7 bits 0-17. Establish absolute references. Establish Extended Memory references. Set buffer size in bits 17-35 of .CRACF+17. Retrieve SCF Trace Buffer size from .MINIT Build trace buffer tally word and store in .CRSTC as buffer control tally word and .CRSTC+2 as tally refresher for .CRSTC. Compute final program size = MSCF size plus trace buffer size and return this value to .MINIT on exit. EXIT		
ESTABLISH SCF DISK I/O CONTROLS	1.1.1	DISKIO
Set data size to 318 words. Set PMX contains Seek Address flag, bit 32, in skeleton word 0. Set absolute address of Seek DCW+1 in skeleton word 5 bits 0-17. Set absolute address of Write DCW in skeleton word 6 bits 0-17. Build Seek DCW in skeleton word 3 and copy to seek DCW in .MSCF. Copy disk file 1 name from ACNMSV (words 0 & 1)in .MINIT to .MSCF. Copy disk file 2 name from ACNMSV words 2 & 3 in .MINIT to .MSCF. EXIT		
ESTABLISH SCF TAPE I/O CONTROLS.	1.1.2	TAPEIO
Set data size to 319 words. Set skeleton word 2 to write tape binary command. Set skeleton word 5 to write tape binary command.		

	MODULE NUMBER	MODULE NAME

```
Set absolute address of Write DCW
   in skeleton to word 17,
   bits 0-17.
Set absolute address of Write DCW+1 in
   skeleton word 4 bits 0-17.
EXIT
```

MASTER MODE ENTRY (MME GESCF) PROCESSOR. 1.2 MMESCF

```
[This enty is reserved for implementation
 in release WW7.1. Entry into .MSCF at this
 location prior to that release will cause the
 caller to abort with a reason code of "AF".]

Set abort code to octal 65,
   'IMPROPER GCOS CALL'.
Go abort slave.
```

SCF RECORD & TRACE PROCESSING 1.3 CALSCF

```
   [This is the entry for .CALL .MSCF,2
    to buffer SCF record for caller.]
Close the SCF gate and inhibit interrupts.
Save callers registers.
If SCF is not configured,
   go to TRACK 1.3.1.
If SCF is not initialized,
   go to TRACK 1.3.1.
If wait for exchange in process,
   go to LOSTLR 1.3.7.
If purge on,
   go to LOSTLR 1.3.7.
Retrieve SCF length and type.
   [Is this SCFAUD?]
   If type = 402 and length = 27,
      perform RECORD 1.3.3.
   [Not SCFAUD]
Perform MAPCK 1.3.5 for recording.
If recording bit for this record is on,
   Perform RECORD 1.3.3.
If SCF type = 1, or 16, or
   (15 and end-of-job detected,
      Perform SYCTR 1.3.2.
```

	MODULE NUMBER	MODULE NAME
SCF TRACE PROCESSING	1.3.1	TRACK

```
    Perform MAPCK 1.3.5 for trace.
    If trace bit for this record is on,
       Perform TRACE 1.3.4.
    Go to EXIT 1.6.
```

SYSTEM COUNTER SCF RECORD.	1.3.2	SYSCTR

```
    Form SCF record 504 in internal buffer.
    Perform MAPCK 1.3.5 for recording.
    If recording bit is on,
       Perform RECORD 1.3.3.
    Perform MAPCK 1.3.5 for trace.
    If trace bit is on,
       Perform TRACE 1.3.4.
    RETURN
```

SCF RECORDING.	1.3.3	RECORD

```
    Retrieve record length.
    Increment record length to include RCW.
       [psuedo SCF RCW and true RCW]
    If record length > words available,
       Perform DOIO 1.3.6
       Go to RECORD 1.3.3.
    Format and store RCW.
    Copy record from caller to SCF buffer.
    Set new buffer position.
    Complete User Header in SCF buffer.
    RETURN.
```

MAKE SCF TRACE TABLE ENTRY.	1.3.4	TRACE

```
    If record will not fit in trace table,
       Reset trace entry from .CSCFT+3.
       Make entry.
    RETURN.
```

	MODULE NUMBER	MODULE NAME
CHECK SELECTED MAP.	1.3.5	MAPCK

```
    [This routine is entered when a check
     of various maps is required. Two exits
     are provided - Exit 1 if bit is on,
     Exit 2 if bit is off. Two parameters are
     furnished - Address of map base, and
     Record Type.]
    Establish map address.
    If bit for Record is on
       Return to Exit 1 (Execute a function).
    If bit for Record is off
       Return to Exit 2 (bypass
          execution of function).
```

SCF I/O CONTROL.	1.3.6	DOIO

```
    If no buffer available,
       go to LOSTLR 1.3.7
    If write is in progress,
       Go to LOSTLR 1.3.7.
    Set write flag to ON.
    Get last BSN from bits 18-35 of .CRACF.
    Increment BSN by 1 and put in SCF buffer
       BCW.
    Build write DCW.
    If tape SCF,
       Copy Write DCW to I/O QUEUE entry.
    Reset stop bit in I/O Queue entry.
    If Extended Memory,
       Reset stop bit in I/O QUEUE
          Table and save caller's MBA.
    Call .MIOS,15 to start I/O.
    if Extended Memory,
       Reload caller's MBA.
Switch SCF buffers in .CRACF+10.
    Zero BCW in new buffer.
    Initialize starting buffer address in
       .CRACF+17.
    RETURN.
```

LOST LOGICAL RECORD CHECK.	1.3.7	LOSTLR

	MODULE NUMBER	MODULE NAME
If caller is not Interupt Handler, Dispatcher, or .MPOPM Go to LOSTIO 1.3.7.1. If no system gates are shut in addition to .CRACF, Go to LOSTIO 1.3.7.1. Increment count of lost variables. Go to EXIT 1.6.		
RELINQUISH CALLER TO PREVENT LOST RECORD	1.3.7.1	LOSTIO
Reload caller's registers. Set I/O Complete since Last Link, bit 27, flag in caller's .STATE word. Relinquish processor for this caller. (Upon return from relinquish) Go to CALSCF 1.3.		
UNIQUE JOB IDENTIFIER MANAGER.	1.4	UJISCF
GENERATE UNIQUE JOB IDENTIFIER.	1.4.1	UJIGEN
<Entry for GEIN and POP6 to obtain a Unique Job Identifier.> Retrieve Unique Identifier from .CRACF+12. Update the Unique Identifier in .CRACF+12. Go to OPEXIT 1.6.1.		
PUT ENTRY IN S.TAB.	1.4.2	UJITBL
(Entry from the Scheduler and POP6) Store SNUMB, Umique Job Identifier and Network Information in SITAB entry corresponding to program number of job. Go to EXIT 1.6.		

	MODULE NUMBER	MODULE NAME
PURGE ENTRY FROM S.TAB	1.4.3	UJIPRG

(this entry is used by the
Peripheral Allocator).
Find entry in S.TAB by SNUMB.
Clear entry to zeroes.
Go to EXIT 1.6.

I/O COURTESY CALL.	1.5	CCSCF

Update BSN in bits 18-35 of .CRACF
by one.
If status return is ready major status
major status with no IOM/channel error,
GOTO GETQ 1.5.1.
<I/O error occurred>
If disk accounting
Go to IOERR
If tape accounting and major status is not
'End of Tape Mark'
Go to IOERR
<Tape EOF>
Call .MACTS,1 to exchange tapes.

WRITE CURRENT ENTRY.	1.5.5.1	WRITEN

Put BSN in record.
Reset exception processing flags in
I/O queue entry.
Link Queue entry first with I/O
stop flag off.
END COURTESY CALL.

GET QUEUE ENTRY.	1.5.1	GETQ

Request Queue Entry.
If none available, relinquish then
Go to GETQ 2.4.
Save I/O Queue entry address in bits 18-35 of
.CRACF+11.
Move Skeleton to Queue entry
Starting at Queue entry word 2.

	MODULE NUMBER	MODULE NAME
Put current buffer address in Write DCW. If tape, Put current buffer address in Queue entry word 5. If tape, Go to NOSEEK 1.5.4 If logical EOF encountered on last write, Go to LOGEOF 1.5.2 Get next Seek address from channel Module. If error return, Go to SEEKER 1.5.3 Retrieve current seek address from .CRETR. Save next seek address in .CRETR. Store current seek address in I/O Queue entry word 4.		
LOGICAL EOF EXCHANGE. CALL to exchange disk files. Go to NOSEEK 1.5.4	1.5.2	LOGEOF
SEEK ADDRESS ERROR. If not logical EOF <Forward to Far> error, Fault <ZOP 5,DU> Set logical EOF encountered flag in .CRACF. Retrieve current seek address from .CRETR. Store seek address in I/O Queue entry. GOTO NOSEEK 1.5.4.	1.5.3	SEEKER
COMMON TAPE/DISK I/O QUEUE ENTRY PROCESSING. Set stop flag, bit 18, in I/O queue entry. Link entry first in I/O queue via .LINKF. Reset SCF Write flag in .CRACF. END COURTESY CALL.	1.5.4	NOSEEK

	MODULE NUMBER	MODULE NAME
I/O ERROR.	1.5.5	IOERR

```
    Set building status in I/O Error Entry.
    If Disk, CALL .MPOPV,2 <Disk error exchange>
       If purge and lose logical record flags on
          in .CRACF,
          GOTO GETQ 1.5.1 <Lose this record>
       Else
          GOTO WRITEN 1.5.5.1.
    If tape, CALL .MACTS,2 <Tape error exchange>.
```

WRITE CURRENT ENTRY.	1.5.5.1	WRITEN

```
    Put BSN in record.
    Reset exception processing flags in
       I/O queue entry.
    Link Queue entry first with I/O
       stop flag off.
    END COURTESY CALL.
```

RESTORE CALLER'S REGISTERS	1.6	EXIT

```
    Restore Caller's registers.
```

OPEN GATE EXIT	1.6.1	OPEXIT

```
    Open .CRACF.
    .EXIT.
```

MODULE NUMBER - 001

MODULE SYMBOL - .MSCF

MODULE NAME - Statistical Collection File Processor

MODULE FUNCTION

The primary functions of this module are:

.MME GESCF processing (MMESCF)
.SCF processing (CALSCF)
.Unique SCF Record Identification Control (UJISCF)
.I/O control via a Courtesy Call

The module will be responsible for all SCF control processing, writing to the SCF file and management of an SCF TRACE Queue.

MODULE REFERENCED BY

023	.MBRT6	273	.MTOLT	215	.MPOP7	406	.MFS31	107	.MGEOT
103	.MGEIN	266	.MSNP3	---	TSSK	414	.MFS37	175	.MIOS
---	TSSI	405	.MFS30	---	TSSL	431	.MFS50	574	.MLOGN
044	.MFALT	777	.MPAAF	433	.MCOLT	432	.MPOLT	434	.MMOLT
450	.MOT12	632	.MFS77	005	.MALC1	315	.MPOQ8	557	.MMOR1
106	.MGENC	217	.MPOP9						

MODULES REFERENCED

004 .MACTS,1
016 .MBRT1,3
033 .MDISP,4 & 5
175 .MIOS,2,4 & 15
004 .MACTS,2
243 .MPOPV,1
243 .MPOPV,2

MODULE NUMBER - 1.1

MODULE SYMBOL - .ISCF

MODULE NAME - SCF Module Initialization

This module establishes references and control values required by GCOS to process SCF records. In addition, absolute references as required by the modules are established and extended memory options are taken into consideration.

The following areas are initialized by .ISCF:

.CRACF+13	Location of the SCF recording map.
.CRACF+14	Location of the SCF disposition map.
.CRACF+15	Location of the operator verb recording map.
.CRACF+16	Location of the SCF trace map.
.CRACF+17	Current Buffer Position
.CRACF+18	Location of SCF Sum map.

Upon establishment of these areas, the map values furnished to .MINIT are obtained from the location furnished by .MINIT in .CRACF+2, bits 18-35. The base for map information is this address plus four. Using this base reference, the maps are moved into .MSCF. The sum map is formed by or'ing together the record and trace maps.

The address of words 2-9 of an I/O entry skeleton within the module is placed in .CRACF+11, bits 0-17, for reference by .MPOPM during assignment of the SCF file.

The I/O skeleton is specialized based on the recording media assigned to the SCF. Disk specialization is accomplished by DISKIO 1.1.2 and Tape specialization by TAPEIO 1.1.3.

Absolute references within the program are established and, if required, extended memory instructions are enabled.

The size of the SCF Trace Buffer is obtained from .MINIT. If the buffer size is > 4095 words, it is reduced to 4095 words. The buffer address and length are stored in .CRSTC+2 and .CRSTC+3. The program size is adjusted by the length of the Trace Buffer and a return is made to .MINIT.
The data length of the SCF buffer is based on the recording media.

INTERNAL REFERENCES

MODULES REFERENCED - DISKIO 1.1.2
TAPEIO 1.1.3

MODULE REFERENCED BY - None

EXTERNAL REFERENCES

MODULES REFERENCED - .MINIT (RETURN)

MODULE REFERENCED BY - 173 .MINIT

REMARKS

The SCF communication region .CRACF is established jointly by .MINIT, .ISCF, and .MPOPM. The global variable list at the end of this section details which areas are processed by each of these modules. The presentation sequence is the processing sequence. Maps of the two .CR areas required for SCF processing follow.

SCF COMMUNICATION REGION

	0	17	18	35
.CRACF+0	CONTROL FLAGS		BLOCK SERIAL NUMBER	
+1	GATE			
+2	SCF PAT WORD 1 (SCT)			
+3	SCF PAT WORD 2 (IOC TIME)			
+4	SCF PAT WORD 3 (CONTROL FLAGS)			
+5	SCF PAT WORD 4 (POSITION)			
+6	SCF PAT WORD 5 (FILE DESCRIPTOR)			
+7	SCF PAT WORD 1 ALTERNATE FILE			
+8	SCF PAT WORD 3 ALTERNATE FILE			
+9	SCF PAT WORD 5 ALTERNATE FILE			
+10	CURRENT BUFFER		ALTERNATE BUFFER	
+11	I/O SKELETON		I/O ENTRY	
+12	UNIQUE JOB IDENTIFIER			
+13	RECORDING MAP (RMAP)		MBZ	
+14	DISPOSITION MAP (DMAP)		MBZ	
+15	VERB MAP (VMAP)		MBZ	
+16	TRACE MAP (TMAP)		MBZ	
+17	OUTPUT BUFFER POSITION		RESERVED	
+18	SUM MAP (SMAP)		MBZ	

MBZ = Must be zeros

SCF TRACE CONTROL REGION

	0	17	18	30	35
.CRSTC+0	TRACE BUFFER POSITION		WORDS REMAINING	0	
+1	GATE				
+2	TRACE BUFFER BASE		BUFFER LENGTH	0	

The map tables generated by this routine are in ten-word groups. Their format and references follow:

MAP REFERENCE	LENGTH	CONTENTS
RMAP	10 WDS	Each word contains recording flags for each SCF class. Bit 0 of each word = 0. Bits are interpreted as follows -

BIT(n)	ACTION
0	Do not record
1	Record

Bits are interpreted positionally and correspond to a record type within a class.

MAP REFERENCE	LENGTH	CONTENTS
DMAP	10 WDS	Same as above except record disposition is represented. Bits are interpreted as follows -

BIT(n)	ACTION
0	Cannot change
1	May change

MAP REFERENCE	LENGTH	CONTENTS
TMAP	10 WDS	Same as above except SCF trace entry is represented. Bits are interpreted as follows -

BIT(n)	ACTION
0	Do not trace
1	Trace

MAP REFERENCE	LENGTH	CONTENTS
VMAP	10 WDS	Each word indicates what operator verbs, by verb class, are to be recorded. Bit zero of the each word is utilized as a disposition indicator. If bit zero = 0, the disposition of all verbs in this class may not be altered by the console operator. All other bits in words 0-9 are representative of a verb or a potential verb. Bits are interpreted as follows -

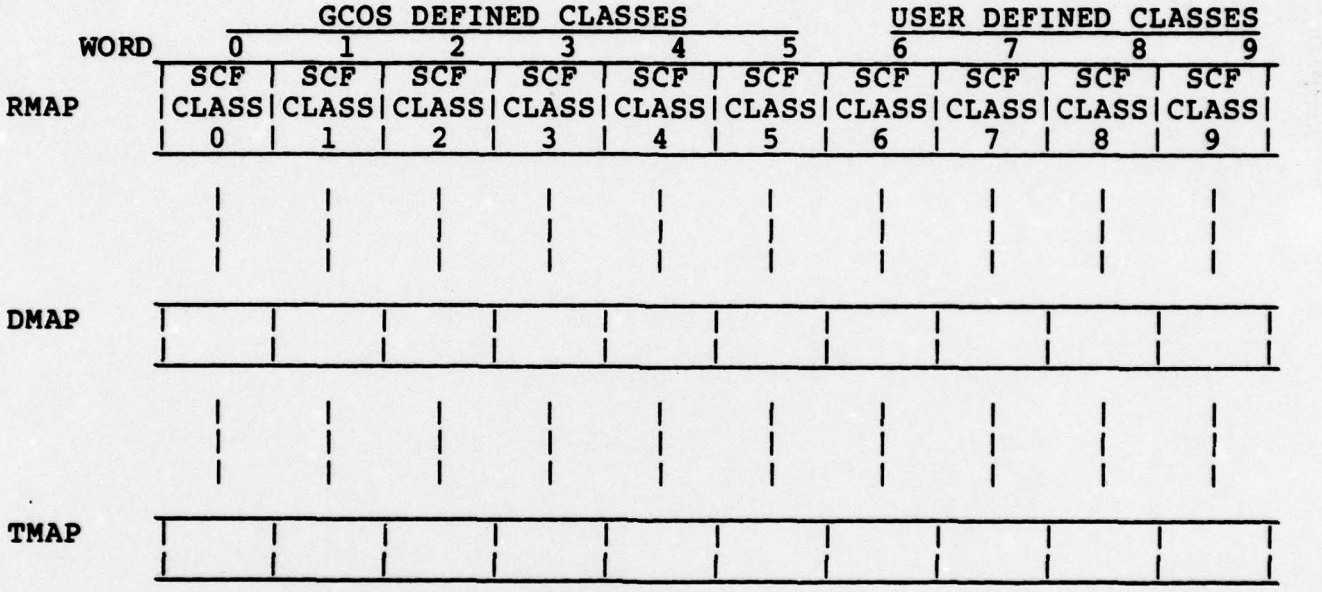
BIT(n)	ACTION
0	Do not journalize
1	Journalize

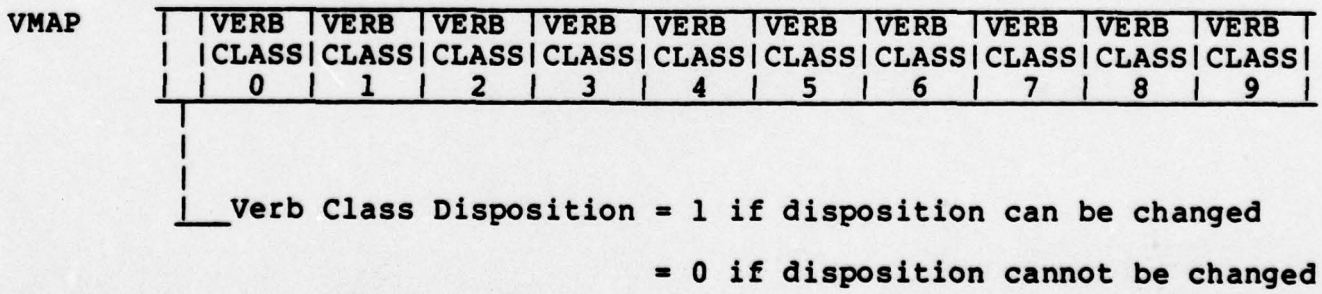
Bit 0 of each word is initialized at startup time from parameters in the startup deck.

NOTE: The disposition can be bypassed by turning off the recording of type 402 SCF records, but the occurrence of turning it off will be recorded if initialized as such.

The following map depicts the referencing between .CRACF regions and the .MSCF maps described above.

.CRACF+13	RMAP	MBZ
+14	DMAP	MBZ
+15	VMAP	MBZ
+16	TMAP	MBZ





The AUDIT verb, which is now renamed to SCFAUD, will be represented in VMAP, but there is no check to VMAP or RMAP when the operator enters the command. The system will prompt the operator for the text as it presently does and will unconditionally write the text to the SCF. Since SCFAUD is a completely voluntary verb, the simple procedure of not using it is the solution for not journalizing the verb.

MODULE NUMBER - 1.1.2

MODULE SYMBOL - DISKIO

MODULE NAME - Establish SCF Disk I/O Controls

MODULE FUNCTION

This module establishes Disk file control information in the I/O Queue entry skeleton when it is determined that the SCF is Disk resident. The following values are established by this routine:

REFERENCE	BIT(S)	CONTENTS
SKELT+3	0-35	Seek DCW
+4	0-17	Absolute address of Seek DCW+1
+6	0-17	Absolute address of Write DCW
FILES+0	0-35	Disk file 1 name (2 words)
FILES+2	0-35	Disk file 2 name (2 words)

Buffer data length is set to 318.

Upon completion of above, the routine continues with common I/O Queue entry processing.

INTERNAL REFERENCES

MODULE REFERENCED - None

MODULE REFERENCED BY - .ISCF 1.1

EXTERNAL REFERENCES - None

MODULE NUMBER - 1.1.3

MODULE SYMBOL - TAPEIO

MODULE NAME - Establish SCF Tape I/O Controls

MODULE FUNCTION

This module establishes tape file control information in the I/O Queue entry skeleton when it is determined that the SCF is tape-resident. The following values are established by this routine:

REFERENCE	BIT(S)	CONTENTS
SKELT+2	0-35	Write Tape Binary command
+4	0-17	Absolute address of Write DCW+1
+5	0-35	Duplicate SKELT+2
+6	0-17	Absolute address of Write DCW

Buffer data length is set to 319.

Upon completion of the above, the routine continues with common I/O Queue entry processing.

INTERNAL REFERENCES

MODULES REFERENCED - None

MODULE REFERENCED BY - .ISCF 1.1

EXTERNAL REFERENCES - None

MODULE NUMBER - 1.2

MODULE SYMBOL - MMESCF

MODULE NAME - Master Mode Entry (MME GESCF) Processor

MODULE FUNCTION

In Release WW6.4, entry into this module will abort the caller with an abort code of octal 65, 'Improper GCOS Call'.

INTERNAL REFERENCES - None

EXTERNAL REFERENCES

MODULES REFERENCED - 016 .MBRT1,3

MODULE REFERENCED BY - None

MODULE NUMBER - 1.3

MODULE SYMBOL - CALSCF (.MSCF,2)

MODULE NAME - SCF Processing and SCF Trace Queue Management

MODULE FUNCTION

Entry into this module is via an external call from a GCOS element. Upon entry, the related modules expect to find the address of an SCF record in index register 1. Checks to determine if a particular SCF record is to be written are made by the calling element by using the .SCFCK macro. Any SCF record presented to the module will be written. If the .SCFCK macro is not utilized by the caller to check the recording status of a given record prior to issuing a call to CALSCF, the record is considered to be one which is unconditionally recorded.

Responsibilities of this module are:

- Saving references to the SCF Record to be processed
- Completion of the SCF Header
- SCF output buffer management
- Output control and error detection
- Unique Identifier queue purge upon detection of Job EOJ status
- SCF trace entry management
- Generation of System Counters record upon detection of job start and job termination

Output control and error detection are accomplished by a courtesy call routine which executes under control of .MPOPM. Output is started by a CALSCF module but again the controlling element is .MPOPM. The SCF output buffers are also located in .MPOPM and are identified by the symbol ACC.AB.

It is possible that a callers SCF record will not be placed in the SCF file. This event will occur under the following conditions:

Disk SCF is configured and
One SCF file is purging and
The alternate SCF file is full

Should the above situation occur, a return will be made to the user .MSCF calling sequence.

A count of lost logical records will be maintained within the module. This count will be recorded on each occurrence of the System Counters record, type 504. The caller will not be informed that the record was lost.

Code within the module is inhibited. The initial sequence within the module performs preliminary tests to determine if an SCF file is configured and available. Failure of this test sequence will result in

a return to the calling element. If the previous tests were successful, the record is a candidate for placement in the SCF buffer (RECORD 1.3.3). Upon return, checks are made to determine if a system counter record is required. Upon detection of job start and job end, this record is formatted by SYSCTR 1.3.2 and RECORD is performed to complete processing. Additional processing at end-of-job is entry into the S.TAB purge routine, QUEPRG 1.3.4, which deletes the terminated SNUMB segment from S.TAB.

Completion of processing results in the restoration of user registers and, if required, Master Base A, the SCF gate is opened and a return is made to the caller via a .EXIT.

INTERNAL REFERENCES

MODULES REFERENCED - EXIT1 1.3.1 SYSCTR 1.3.3 MAPCK 2.2
RECORD 1.3.3 QUEPRG 1.3.4 TRACE 2.3
MEXIT 1.3.2

MODULE REFERENCED BY - None

EXTERNAL REFERENCES

MODULES REFERENCED - See .MSCF 001

MODULE REFERENCED BY - See .MSCF 001

SYSTEM GATES REFERENCED - .CRACF
.CRSTC

COMMUNICATIONS REGIONS REFERENCED

.CRACF .CRSTC .CRETR

CALLING SEQUENCE

```
EAX1 (address of callers SCF record)
.CALL .MSCF,2
```

MODULE NUMBER - 1.3.1

MODULE SYMBOL - EXIT

MODULE NAME - Restore Callers Registers

MODULE FUNCTION

Restore caller's registers and master base A prior to returning to caller. The SCF gate is opened and the return is made.

INTERNAL REFERENCES

MODULES REFERENCED - None

MODULE REFERENCED BY - CALSCF 1.3

EXTERNAL REFERENCES - None

MODULES REFERENCED

MODULE NUMBER - 1.3.2

MODULE SYMBOL - MEXIT

MODULE NAME - Return to .MSCF Call

MODULE FUNCTION

In the event that the alternate SCF is purging and the current SCF is full, control is returned to the user calling sequence. In addition, if no SCF file is available (file exchange in progress), the same return is made to the user. This module restores the users registers, opens the SCF gate and returns via an .EXIT-2.

INTERNAL REFERENCES

MODULES REFERENCED - None

MODULE REFERENCED BY - CALSCF 1.3 DOIO 3.1
LOSTLR 3.1.2

MODULE NUMBER - 1.3.3

MODULE SYMBOL - SYSCTR

MODULE NAME - System Counter SCF Record Type 502

MODULE FUNCTION

Upon detection of an SCF record type 001 and activity 1, or 015 and end-of-job, control is given to this module to format a system counter SCF record. If the recording indicator is on, the location of the record is placed in X1 and control is given to RECORD 1.3.3, otherwise, a return to CALSCF 1.3 is made.

SCF RECORD FORMAT

WORD	BIT(S)	DESCRIPTION
0-13		STANDARD HEADER
14-17		.CRIDT - PROCESSOR IDLE TIME
18		.CROUH - ACCUMULATED OVERHEAD TIME
19		.CRTAL - TOTAL ACTIVITIES
20		.CRTCN - TOTAL CONNECTS
21		.CRIOS - TOTAL DISPATCHES
22		.CRTIR - TOTAL INTERRUPTS
23		.CRTJB - TOTAL JOBS SINCE STARTUP
24		.CRTJM - TOTAL JOBS MOVED
25		.CRTJS - TOTAL JOBS SWAPPED
26		.CRTLS - TOTAL LOST INTERRUPTS
27		.CRTRJ - TOTAL REMOTE JOBS
28		.CRTRR - TOTAL REMOTE REPORTS
29		.CCTSN - TOTAL CARDS PUNCHED BY SYSOUT
30		.CRTSR - TOTAL LINES PRINTED BY SYSOUT
31		.CRTWT - TOTAL TIMES PROCESSOR IDLE
32		LOSTLR - LOST SCF RECORDS

INTERNAL REFERENCES

MODULES REFERENCED - RECORD 1.3.3 MPCK 2.2 TRACE 2.3

MODULE REFERENCED BY - CALSCF 1.3

EXTERNAL REFERENCES - None

MODULE NUMBER - 1.3.4

MODULE SYMBOL - QUEPRG

MODULE NAME - Unique ID Queue Purge

MODULE FUNCTION

This module is entered from CALSCF 1.3 upon detection of a type 015 record and end-of-job has been found. The entry for the SNUMB in the record is traced in the Unique ID table, S.TAB. Once the SNUMB has been located, the table segment is set to spaces and the table entry count is reduced by 1. A return to CALSCF is made.

INTERNAL REFERENCES

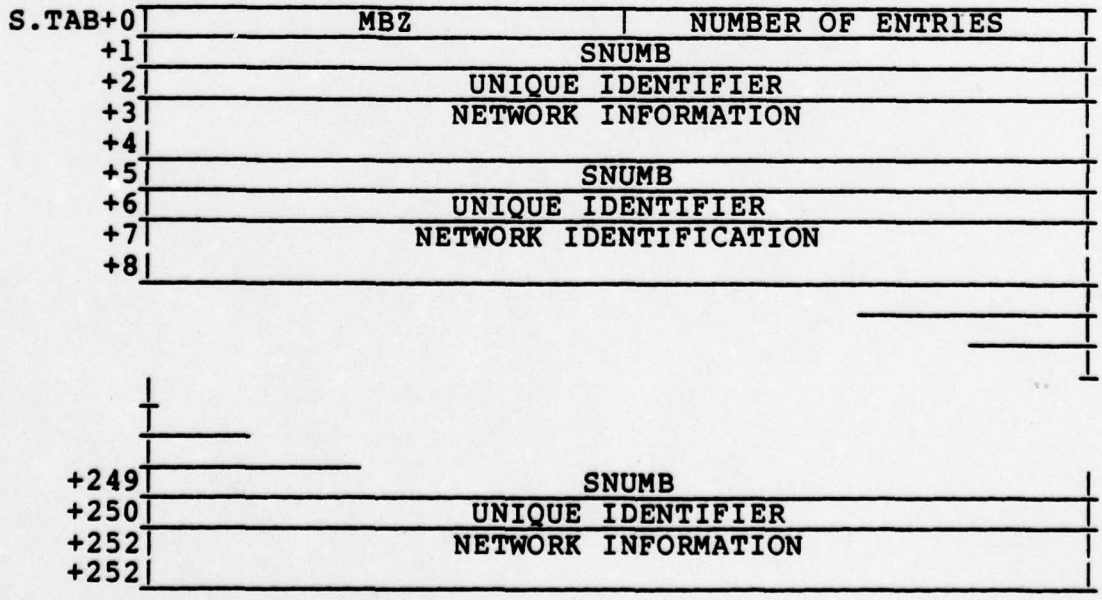
MODULES REFERENCED - None

MODULE REFERENCED BY - CALSCF 1.3
SCHED2 1.4.3

EXTERNAL REFERENCES - None

REMARKS

The S.TAB table entries are made by the Unique Identifier Queue Manager, UJISCF 1.4. The Network Information relative to a given SNUMB also appears within the table. The initial value of S.TAB is all zeros. Purged entries are reset to zeros. Position zero is the S.TAB control word. The first 15 segments of S.TAB are reserved for system programs. The format of S.TAB appears below:



11/21/77

SNUMB

Established by the program start function, .MGEIN or .MPOP5 by calling .MSCF,3.

UNIQUE IDENTIFIER - Obtained from .CRACF+12 by .MSCF,3

NETWORK INFORMATION - Sources are PWIN and, in release WW7.1, RNP log-on.

PWIN - The Identification of the User Host

RNP(WW7.1) - The RNP User Identification

MODULE NUMBER - 1.3.3

MODULE SYMBOL - RECORD

MODULE NAME - Process SCF Record

MODULE FUNCTION

The record length is retrieved from the user buffer and incremented by 1 to include the RCW. A test is made to determine if the record will fit into the SCF buffer. If the record will not fit, DOIO 3.1 is performed and upon return, control is passed to the beginning of the module. Otherwise, the RCW is placed into the SCF buffer, the SCF Header is completed by filling in:

- User Host System ID from S.TAB
- Server Host System ID from .CRSID
- Year, Month, Day from .CRDAT
- Creation time from .CRDAT+1
- Unique Identifier from S.TAB
- The conditional entries (see below) SNUMB and USERID are completed if their storage areas contain zeros.

The record is then stored in the SCF buffer. Prior to exit from the module, a test is made to determine if enough space remains to enter the smallest size SCF record (16 words) into the SCF buffer a negative test result will cause DOIO 3.1 to be executed. Upon return from DOIO, a return to the module caller is initiated.

INTERNAL REFERENCES

MODULES REFERENCED - DOIO 1.3.1

MODULE REFERENCED BY - CALSCF 1.3
SYSCTR 1.3.5

EXTERNAL REFERENCES - None

MODULE NUMBER - 2.2

MODULE SYMBOL - MAPCK

MODULE NAME - Check Selected Map

MODULE FUNCTION

This routine is entered to test for the presence of a specific bit in any of the SCF status maps. Parameters required by this routine are the SCF record type and the .CRACF+n location of the map base. Two returns to the calling module are possible.

1. End of calling sequence + 1 = The selected bit is on.

2. End of calling sequence + 2 = The selected bit is off.

INTERNAL REFERENCES

MODULES REFERENCED - None

MODULE REFERENCED BY - CALSCF 1.3
SYSCTR 1.3.3

EXTERNAL REFERENCES - None

CALLING SEQUENCE

```
LDA     0,1     Record Type in lower
TSX0    MAPCK
ZERO    .CRACF+13,0 The recording map is checked for type in AL
....    Return 1 - Record type bit is = 1
....    Return 2 - Record type bit is = 0
```

The verb map, .CRACF+15 cannot be properly tested by this routine.

REMARKS

Space for the Standard Header, excluding the RCW must be allowed for within the user buffer. The expected format of the Standard Header excluding the RCW is as follows:

STANDARD SCF HEADER

WORD	BIT(S)	DESCRIPTION	ENTERED BY
1	0-17	RECORD SIZE	FORMATTING MODULE
	18-20	CONTINUATION INDICATOR = 000 - NO CONTINUATION = 010 - A CONTINUATION RECORD WILL FOLLOW = 100 - INTERMEDIATE CONTINUATION RECORD = 110 - LAST CONTINUATION RECORD	FORMATTING MODULE
	21-35	RECORD TYPE	FORMATTING MODULE
2- 3		SYSTEM IDENTIFICATION (USER HOST)	.MSCF
4- 5		SYSTEM IDENTIFICATION (SERVER HOST)	.MSCF
6		RECORD CREATION DATE (YYMMDD)	.MSCF
7		RECORD CREATION TIME (CLOCK PULSES)	.MSCF
8	0-29	SNUMB	.MSCF *
	30-35	VARIABLE, DEPENDING ON RECORD TYPE	.MSCF *
9		UNIQUE IDENTIFIER	.MSCF
10-11		USERID	.MSCF *
12	0-11	MBZ	.MSCF *
	12-17	PROGRAM NUMBER	.MSCF *
	18-35	ACTIVITY NUMBER AS -nn	.MSCF *
13	0- 5	FNP NUMBER	FORMATTING MODULE
	18-23	TERMINAL TYPE	FORMATTING MODULE
	24-35	TERMINAL IDENTIFICATION	FORMATTING MODULE

Words 2 thru 7 will be completed by RECORD 1.3.3. The remaining words must be completed by the caller. The * indicates values which are not always available to .MSCF and must be entered by the calling module. Specifically identified are GEIN, TSS and FILSYS actions for TSS users. TSS areas complete only the USERID. GEIN must complete the SNUMB, the USERID, and the Unique Identifier.

It is the callers responsibility to ensure that words in the Standard SCF Header that are not applicable to a specific record or an instance of a given record type are zero filled.

The SCF buffer in .MPOPM will be 320 words in length. For purpose of control, allowance must be made for at least 1 BCW, 1 RCW, 1 SCF record of 317 words and a possible logical EOF indicator.

MODULE NUMBER - 2.3

MODULE SYMBOL - TRACE

MODULE NAME - Make SCF Trace Entry

MODULE FUNCTION

This module is entered from CALSCF 1.3 when it has been determined that a Trace Table entry is required. The record length is compared to the current delta value in .CRSTC+2. If the record will fit, it is placed in the Trace Table and a return to CALSCF is made. Otherwise, .CRSTC+2 is reinitialized from .CRSTC+3 and control is given to the beginning of TRACE.

INTERNAL REFERENCES

MODULES REFERENCED - None

MODULE REFERENCED BY - CALSCF 1.3 SYSCTR 1.3.3

EXTERNAL REFERENCES - None

REMARKS

The SCF Trace Table is a circular buffer into which selected SCF records are placed Entries are made sequentially into the buffer until the buffer is full. When it is determined that the buffer is full, references to the buffer are reestablished to the buffers start point. Both .CRSTC and the Trace Table location and length are established by system startup.

MODULE NUMBER - 3.1

MODULE SYMBOL - DOIO

MODULE NAME - SCF I/O Control

MODULE FUNCTION

This module determines if I/O can be performed. If a write is in progress, .CRACF bit 0 = 1, LOSTLR is invoked to determine if the callers record content will be lost. Otherwise, the write flag is turned ON and processing continues with DOIO1.

INTERNAL REFERENCES

MODULES REFERENCED - DOIO1 3.1.1 (Implicit)
LOSTLR 3.1.2

MODULE REFERENCED BY - RECORD 1.3.3

EXTERNAL REFERENCES - None

MODULE NUMBER - 3.1.1

MODULE SYMBOL - DOIO1

MODULE NAME - Prepare for I/O (Entry 1)

MODULE FUNCTION

The Block Serial Number (BSN) is obtained from .CRACF, bits 18-35 and placed in the Block Control Word (BCW), bits 0-17. The BSN in the BCW is incremented by +1. If the SCF is assigned to tape, the record length is stored in the I/O Queue entry. The word count is then stored in the write DCW. The buffer length - 1 is then stored in the BCW, bits 18-35. Processing continues with DOIO2.

INTERNAL REFERENCES

MODULES REFERENCED - DOIO2 (Implicit) 1.3.1.1

MODULE REFERENCED BY - DOIO (Implicit) 3.1

EXTERNAL REFERENCES - None

MODULE NUMBER - 3.1.1.1

MODULE SYMBOL - DOIO2

MODULE NAME - Prepare for I/O (Entry 2)

MODULE FUNCTION

DOIO2 is a continuation of preparation for I/O. The routine establishes an I/O trace entry and then resets the I/O stop bit in the I/O Queue entry, places the logical channel index in index register 2 and calls entry point 15 of IOS (Start I/O). Upon return, the output buffer control tally word is reestablished with the alternate buffer address and length. The address values in .CRACF+10 bits 0-17 and .CRACF+10, bits 18-35 are exchanged. A full buffer request results in a return to the caller. I/O is executed under control of program 1.

INTERNAL REFERENCES

MODULES REFERENCED - None

MODULE REFERENCED BY - DOIO1 (Implicit) 3.1.1

EXTERNAL REFERENCES

MODULES REFERENCED - 175 .MIOS,15

MODULE REFERENCED BY - None

MODULE NUMBER - 3.1.2

MODULE SYMBOL - LOSTLR

MODULE NAME - Lost Logical Record Check

MODULE FUNCTION

This module is invoked when a user record is available for recording, the current buffer is full and a write of the alternate buffer is in progress. If the call is the Interrupt Handler, the record is counted as lost, registers are restored and return is made via EXIT1. Otherwise registers are restored, the SCF gate is opened, a return is made to the user .CALL for reissue via .EXIT-2.

INTERNAL REFERENCES

MODULES REFERENCED - MEXIT 1.3.2

MODULE REFERENCED BY - DOIO 3.1

EXTERNAL REFERENCES - None

MODULE NUMBER - 1.4

MODULE SYMBOL - UJISCF

MODULE NAME - Unique Identifier Manager

MODULE FUNCTION

This module has three entry points which function as follows:

1.4.1 GEINID - This entry is used to obtain a unique identifier.

1.4.2 SCHED1 - This entry is used to make entries in the S.TAB table.

1.4.3 SCHED2 - This entry is used to remove S.TAB table entries if errors are discovered during job preparation.

The above named modules will be defined as GCOS entry points.

INTERNAL REFERENCES - None

EXTERNAL REFERENCES

MODULES REFERENCED - None

MODULES REFERENCED BY - 103 .MGEIN
256 .MRGIN
214 .MPOP6
005 .MALC1

MODULE NUMBER - 1.4.1

MODULE SYMBOL - GEINID

MODULE NAME - Retrieve Current Unique ID

MODULE FUNCTION

This module is entered from .MGEIN upon detection of a SNUMB record. Upon entry, the .CRACF gate is closed and the current Unique Identifier is retrieved from .CRACF+12 and placed into the A register for return to .MGEIN. .CRACF+12 is incremented by 1, the SCF gate is opened and a return is made to .MGEIN.

This module is also entered when a system program is started. The final module in the program start sequence is .MPOP5. However, this module is at maximum size and a module with adequate space has not yet been located. If a module with adequate space is not found prior to implementation, SCF requirements for system program start will be defined in a new module.

INTERNAL REFERENCES - None

EXTERNAL REFERENCES

MODULES REFERENCED - None

MODULE REFERENCED BY - 103 .MGEIN
214 .MPOP6

CALLING SEQUENCE

```
.CALL .MSCF,3
```

Upon return to the caller, the current unique identifier will be found in the A register.

GATE REFERENCES

This module closes the SCF gate while manipulating the Unique Identifier.

MODULE NUMBER - 1.4.2

MODULE SYMBOL - SCHED1

MODULE NAME - Place Entry in S.TAB

MODULE FUNCTION

When a job is ready for allocation, the scheduler enters this module to table control values. Upon entry into the module, a search of S.TAB is made to find an available table segment. Upon locating an available segment, the Unique Identifier, SNUMB and Network Identification Information are placed in this table segment and a return is made to the caller. The number of entries in S.TAB is incremented by 1.

This module is also entered when a system program is being started. Entry is after a call is made to GEINID to obtain a Unique Identifier.

INTERNAL REFERENCES - None

EXTERNAL REFERENCES

MODULES REFERENCED - None

MODULE REFERENCED BY - 256 .MRGIN
214 .MPOP6

CALLING SEQUENCE

```
SCHEDULER
    EAX1    (Address of $SNUMB record)
    .CALL   .MSCF,4


.MPOP6
    LDA     (Unique Identifier)
    LDQ     (SNUMB)
    .CALL    .MSCF,4
```

MODULE NUMBER - 1.4.3

MODULE SYMBOL - SCHED2

MODULE NAME - Error, Purge SNUMB from S.TAB

MODULE FUNCTION

This module is entered when a GEIN error is discovered by the Scheduler or if a job is deleted by the Peripheral Allocator. The S.TAB entry is removed from the table by performing the Purge S.TAB routine. Return from the S.TAB purge results in a return to the caller.

INTERNAL REFERENCES

MODULES REFERENCED - QUEPRG 1.3.4

MODULE REFERENCED BY - None

EXTERNAL REFERENCES

MODULES REFERENCED - None

MODULE REFERENCED BY - 256 .MRGIN
005 .MALC1

CALLING SEQUENCE

```
LDA     (SNUMB)
.CALL   .MSCF,5
```

MODULE NUMBER - 1.5

MODULE SYMBOL - CCSCF

MODULE NAME - SCF I/O Courtesy Call

MODULE FUNCTION

This module is executed under control of program 1 upon completion of SCF I/O. Upon entry into the module a test is made to determine if a file exchange request has been made. If this test is true, control is given to DOT 1.5.1. Otherwise, the block serial number field is updated and the status of the last I/O for SCF is checked. The following are included in the status test:

1. Check for device abort Status - if positive result, pass control to I/O Error processing IOER 1.5.2.

2. Check for Channel ready - if positive result, pass control to Get an I/O Queue Entry GETQ 2.4.

3. Check if write flag on in .CRACF - if positive result, pass control to IOER 1.5.2.

4. Check channel busy, devices busy, or alert status - if positive result, pass control to I/O Error processing IOER 1.5.2.

Control passes to the next module if all tests are false.

INTERNAL REFERENCES

MODULES REFERENCED - DOT 1.5.1
GETQ 2.4
IOER 1.5.2

MODULE REFERENCED BY - None

EXTERNAL REFERENCES - None

SYSTEM GATES REFERENCED - .CRACF

COMMUNICATION REGIONS REFERENCED - .CRACF .CRETR

MODULE NUMBER - 1.5.1

MODULE SYMBOL - DOT

MODULE NAME - Disk/Tape Exchange

MODULE FUNCTION

Entry into this module will cause a disk/tape file exchange to occur. A test is made to determine if the SCF is configured on disk or tape. If tape is configured, .MACTS,1 is called to initiate exchange procedures. Otherwise, the disk EOF indicator is set in .CRACF and .MPOPV,1 is called to initiate exchange procedures. Upon return to this module, control is passed to the next module.

INTERNAL REFERENCES

MODULES REFERENCED - GETQ 2.4

MODULE REFERENCED BY - CCSCF 1.5

EXTERNAL REFERENCES

MODULES REFERENCED - 004 .MACTS (Tape SCF)
242 .MPOPU (Implicit)
243 .MPOPV (Disk SCF)
244 .MPOPW (Implicit)
201 .MLBL4 (Implicit)
SCF Purge Utility (Implicit)

MODULE REFERENCED BY - None

MODULE NUMBER - 1.5.2

MODULE SYMBOL - IOER

MODULE NAME - I/O Error

MODULE FUNCTION

In the event that an I/O error is encountered, entry into this module initiates file exchange functions. The I/O Queue entry state is set to building. If the SCF recording media is tape, .MACTS,2 is invoked to initiate a tape file exchange. Otherwise, .MPOPV,2 is invoked to initiate a disk file exchange.

Upon return from the file exchange function, if disk accounting is configured, a test is made to determine if the purge switch in .CRACF is ON. If the test is not true, control is passed to GETQ 2.4. Otherwise, control is given to the next module.

INTERNAL REFERENCES

MODULES REFERENCED - GETQ 2.4

MODULE REFERENCED BY - CCSCF 1.5

EXTERNAL REFERENCES

MODULES REFERENCED - 004 .MACTS (Tape SCF)
242 .MPOPU (Implicit)
243 .MPOPV (Disk SCF)
244 .MPOPW (Implicit)
201 .MLBL4 (Implicit)
SCF Purge Utility

MODULE REFERENCED BY - None

MODULE NUMBER - 1.5.2.1

MODULE SYMBOL - WRITEN

MODULE NAME - Write Current Entry (Error Record)

MODULE FUNCTION

The BSN is placed in the current buffer at word 1. The EOT and Exception Processing flags are reset in the I/O Queue entry and, the queue entry is linked first with the I/O stop bit off and the courtesy call is terminated.

INTERNAL REFERENCES

MODULES REFERENCED - None

MODULE REFERENCED BY - IOER 1.5.2

EXTERNAL REFERENCES - None

MODULE NUMBER - 2.4

MODULE SYMBOL - GETQ

MODULE NAME - Get an I/O Queue Entry

MODULE FUNCTION

A request for an I/O Queue entry, .QUEUE is issued. If no queue is available, a loop to the request is executed until one is obtained. The queue entry address is saved in bits 18-35 of CRACF+11. The I/O entry skeleton, generated by Startup, is moved to the I/O Queue entry. If recording media is tape, the data buffer address is in the I/O Queue entry. The data buffer address is stored in the DCW word.

If recording media is tape, control is passed to NOSEEK 3.2. If the EOF switch is on in .CRACF, control is given to LOGEOF 2.4.1.

Request the next seek address from the Channel module. If an error return is made, control is passed to SEEKER 2.4.2. Otherwise, save the next seek and store the current seek address in the I/O Queue entry. Control is given to the next module.

INTERNAL REFERENCES

MODULES REFERENCED - NOSEEK 3.2
LOGEOF 2.4.1
SEEKER 2.4.2

MODULE REFERENCED BY - DOT 1.5.2 IOER 1.5.2

EXTERNAL REFERENCES

MODULES REFERENCED - Channel Module for Seek Address

MODULE REFERENCED BY - None

MODULE NUMBER - 2.4.1

MODULE SYMBOL - LOGEOF

MODULE NAME - Logical EOF

MODULE FUNCTION

This module is entered when logical EOF is encountered on disk. The file exchange is initiated by a call to .MPOPV,1. Upon return, the I/O Queue entry address is retrieved from .CRACF+11, bits 18-35 and control is given to NOSEEK 3.2.

INTERNAL REFERENCES

MODULES REFERENCED - NOSEEK 3.2

MODULE REFERENCED BY - GETQ 2.4

EXTERNAL REFERENCES

MODULES REFERENCED - 243 .MPOPV (Disk SCF)
244 .MPOPW
SCF Purge Utility

MODULE NUMBER - 2.4.2

MODULE SYMBOL - SEEKER

MODULE NAME - Seek Error

MODULE FUNCTION

This module is invoked via the error return from the channel module if a seek address is not available. Upon entry into the module, status is checked to determine if the error reason is logical EOF encountered. If so, the Logical EOF flag is set in .CRACF. The available seek address is retrieved from .CRETR and placed in the I/O Queue Entry and control is given to NOSEEK 1.5.4. If the error reason is other than EOF encountered, the module will fault with ZOP 5,DU.

INTERNAL REFERENCES

MODULES REFERENCED - NOSEEK 3.2

MODULE REFERENCED BY - GETQ 2.4

EXTERNAL REFERENCES

MODULES REFERENCED - 044 .MFALT (Indirect, via ZOP)

MODULE NUMBER - 3.2

MODULE SYMBOL - NOSEEK

MODULE NAME - Common Tape/Disk I/O Queue Entry Processing

MODULE FUNCTION

Entry to this module completes the current I/O Queue entry. The I/O stop bit is set on in the queue entry and the entry is linked first in the I/O Queue. The SCF Write flag in .CRACF is turned off and the courtesy call is terminated.

INTERNAL REFERENCES

MODULES REFERENCED - None

MODULE REFERENCED BY - GETQ 2.4
LOGEOF 2.4.1
SEEKER 2.4.2

EXTERNAL REFERENCES - None

GLOBAL VARIABLES REFERENCED BY .MSCF & ESTABLISHED BY .MINIT

VARIABLE NAME	INITIALIZED BY:	WHERE DEFINED	DESCRIPTION
ACNMSV	.MINIT	.MINIT	4 WORDS - Alphanumeric. If disk accounting is configured, contains the names of the two SCF files. File 1 name ACNMSV,+1 File 2 name ACNMSV+2,+3 If tape accounting is configured: Tape name ACNMSV+3 Characters 3-5
RMAP	.MINIT	.MINIT	10 WORDS - Binary Each word represents an SCF class. Each bit within a word represents the recording status of a record within a class. Bit 0 in all words must be zero. Bit(n) Value — Condition 0 — Do not record 1 — Record

```
                                .MINIT
                                 RMAP
                         11111111|112222222222333333
bit - 012345678901234567|890123456789012345
      011111000000000100|000000000000000000
```

The above represents the bit pattern of word 0 of RMAP if:

$ SCFBUF RECORD/1,2,3,4,5,15/

NOTE: See .MINIT modifications for formats of $SCFBUF control card.

11/21/77

GLOBAL VARIABLES REFERENCED BY .MSCF & ESTABLISHED BY .MINIT

VARIABLE NAME	INITIALIZED BY:	WHERE DEFINED	DESCRIPTION
DMAP	.MINIT	.MINIT	10 WORDS - Binary Similar to RMAP except element controlled is SCF record recording mode changes. Bit(n) Value — Condition 0 — Cannot alter initial mode 1 — Initial mode may be altered
TMAP	.MINIT	.MINIT	10 WORDS - Binary Similar to RMAP except element controlled is SCF record trace. Bit(n) Value — Condition 0 — Do not trace 1 — Make trace entry
VMAP	.MINIT	.MINIT	10 WORDS - Binary Similar to RMAP except that the element controlled is journalization of operator verbs. Bit(n) Value — Condition 0 — Do not record verb(n) 1 — Verb activity is recorded
TRSIZE	.MINIT	.MINIT	1 WORD - Numeric (Binary) Size of SCF Trace Buffer

The above areas are defined contiguously within .MINIT using ACNMSV as a base reference.

GLOBAL VARIABLES REFERENCED BY .MSCF & ESTABLISHED BY .MINIT

VARIABLE NAME	INITIALIZED BY:	WHERE DEFINED	DESCRIPTION
.CRACF	.MINIT	GCOS Common	18 WORDS - Variable .CRACF defines the base word of the GCOS communication region in which SCF control elements are stored. Areas within .CRACF are addressed relative to this base. The following are established by .MINIT.

Word	Bit(s)	Contents	SCF FILE DEVICE Disk	Tape
0	0	WRITING FLAG	0	0
	1	NOT INITIALIZED	1	1
	2	LOGICAL EOF ENCOUNTERED	0	0
	3	TAPE DEFINED	0	1
	4	DISK FILE 1 DEFINED	1	0
	5	DISK FILE 2 DEFINED	1	0
	6	CURRENT FILE IS TAPE	0	1
	7	CURRENT FILE IS DISK FILE 2	0	0
	8	CURRENT FILE IS DISK FILE 1	1	0
	9-16	ZEROS		
	17	= 1 if I-D-S CONFIGURED		
	18-35	ZEROS		
1				
2	0-17	PAT WORD 1 (SCT POINTER)		
	18-35	POINTER TO ACNMSV FOR USE BY .ISCF		
3	0-35	PAT WORD 2 (IOC TIME), SET TO ZERO		
4	0-35	PAT WORD 3 (CONTROL FLAGS), NOT ESTABLISHED		
5	0-35	PAT WORD 4 (POSITION)		
6	0-35	PAT WORD 5 (FILE DESCRIPTOR)		
7	0-17	ALTERNATE FILE PAT WORD 1		
8	0-35	ALTERNATE FILE PAT WORD 3, NOT ESTABLISHED		
9	0-35	ALTERNATE FILE PAT WORD 5		
10	0-35	ZEROS		
11	0-17	POINTER TO I/O ENTRY SKELETON (SET TO ZERO)		
	18-35	ZEROS		
12	0-35	UNIQUE ID (SET TO 1)		
13-17		SET TO ZEROS		

GLOBAL VARIABLES REFERENCED AND ESTABLISHED BY .MSCF(.ISCF)

VARIABLE NAME	INITIALIZED BY:	WHERE DEFINED	DESCRIPTION
.CRACF	.MSCF	GCOS Common	18 WORDS - Variable .MSCF(.ISCF) continues initialization of this area appending its values to those set by .MINIT.

Word	Bit(s)	Contents
0-11		AS PREVIOUSLY ESTABLISHED
12	0-17	POINTER TO I/O ENTRY SKELETON (.MSCF)
	18-35	UNIQUE ID (.MINIT)
13	0-17	POINTER TO RECORDING MAP BASE
	18-35	MUST BE ZERO
14	0-17	POINTER TO DISPOSITION MAP BASE
	18-35	MUST BE ZERO
15	0-17	POINTER TO VERB RECORDING MAP BASE
	18-35	MUST BE ZERO
16	0-17	POINTER TO SCF TRACE MAP BASE
	18-35	MUST BE ZERO
17	0-35	OUTPUT TALLY

.CRSTC	.MSCF	GCOS Common	3 WORDS - Variable This area is utilized for trace table control.

WORD	BITS	CONTENTS
0	0-17	MBA OF TRACE TABLE. (ZERO IN 6.4)
	18-35	RESERVED FOR GCOS
1	0-35	GATE
2	0-17	ADDRESS OF NEXT INPUT
	18-29	TALLY
	30-35	DELTA (SET TO 0)
3	0-17	BASE ADDRESS OF SCF TRACE TABLE
	18-29	TALLY
	30-35	DELTA (SET TO 0)

GLOBAL VARIABLES REFERENCED BY .MSCF & ESTABLISHED BY .MPOPM

VARIABLE NAME	INITIALIZED BY:	WHERE DEFINED	DESCRIPTION
ACC.AB	.MPOPM	.MPOPM	640 WORDS - SCF I/O Buffer
.CRACF	.MPOPM	GCOS Common	18 WORDS - Variable POPM continues initialization of this area appending its values to those set by .MINIT and .MSCF.

WORD	BITS	CONTENTS
0	0	WRITE IN PROCESS FLAG = 0
	1	NOT INITIALIZED FLAG (.MPOPM SETS TO 1)
	2	LOGICAL EOF FOUND = 0
	3-10	ESTABLISHED BY .MINIT
	11-15	ZERO (RESERVED)
	16	BTL WRITTEN (.MPOPM SETS TO 1)
	17	ZERO (RESERVED)
	18-35	CURRENT BLOCK SERIAL NUMBER
1	0-35	GATE
2	0-17	PAT WORD 1 (SCT) (.MINIT)
	18-35	POINTER TO ACNMSV IN .MINIT
3		PAT WORD 2 (IOC TIME)
4		PAT WORD 3 (CONTROL FLAGE) (.MPOPM)
5		PAT WORD 4 (POSITION) (.MPOPM)
6		PAT WORD 5 (FILE DESCRIPTOR) (.MPOPM)
7		ALTERNATE FILE PAT WORD 1 (.MINIT)
8		ALTERNATE FILE PAT WORD 3 (.MPOPM)
9		ALTERNATE FILE PAT WORD 5 (.MPOPM)
10	0-17	CURRENT BUFFER POINTER (.MPOPM)
	18-35	ALTERNATE BUFFER POINTER (.MPOPM)
11	0-17	POINTER TO I/O ENTRY SKELETON (.MPOPM)
	18-35	CURRENT I/O ENTRY (.MPOPM)
12	0-35	UNIQUE ID (.MINIT, .MPOPM)
13-17		AS ESTABLISHED BY .MSCF

.CRETR	.MPOPM	GCOS Common	1 WORD - Binary Next SCF Seek Address

SECTION 5

PAAF HFD

.MPAAF HIERARCHICAL FUNCTION DIAGRAM

	MODULE NUMBER	MODULE NAME
PERIPHERAL ALLOCATION ACCOUNTING RECORD FORMATION	1.0	.MPAAF
ADDITIONAL CORE REQUEST	1.1	MORCOR
ADDITIONAL MASS STORAGE/TAPE REQUEST	1.2	MORFIL
RELEASE CORE REQUEST	1.3	RELCOR
RELEASE MASS STORAGE/TAPE REQUEST	1.4	RELFIL
FORMAT RECORD HEADER	1.5	HEADER

ADDITIONAL CORE REQUEST 1.1 MORCOR

Move additional number of blocks of core
from .STEMP+8 of the requesting
programs Slave Service Area to the
SCF record.
Save PGM # and LAL of slave being
serviced and file in appropriate
words in header.
Go to 1.5 Header.

	MODULE NUMBER	MODULE NAME
ADDITIONAL MASS STORAGE/TAPE REQUEST	1.2	MORFIL

Get the Peripheral Assignment Table (PAT) pointer for the additional file from the requesting programs .STEMP+10.
Extract the following information from the PAT pointer (.SPATP), PAT body, Secondary SCT and .CRCT1 and move it to the SCF record.
If device not mass storage, Go to 1.2.2 MORTAP.

INFORMATION	FROM	TO
DEVICE TYPE	SCT	WORD 14 OF RECORD
DISPOSITION CODE	PAT POINTER	WORD 14 OF RECORD
FILE CODE	PAT POINTER	WORD 14 OF RECORD
IOM NUMBER	.CRCT1	WORD 18 OF RECORD
CHANNEL NUMBER	.CRCT1	WORD 18 OF RECORD
DEVICE NUMBER	SCT	WORD 18 OF RECORD

	MODULE NUMBER	MODULE NAME
PROCESS MASS STORAGE REQUEST	1.2.1	MORMAS

Extract information from the PAT for the appropriate mass storage file using the PAT obtained in 1.2 MORFIL.
Go to 1.5 Header.

INFORMATION	FROM	TO
INITIAL SIZE UNAVAILABLE FOR DISK	PAT BODY	SCF RECORD WORD 14
INITIAL SIZE IN LLINKS	PAT BODY	SCF RECORD WORD 15
INITIAL RELATIVE LLINK POSITION	PAT BODY	SCF RECORD WORD 16
MAXIMUM SIZE IN LLINKS	PAT BODY	SCF RECORD WORD 17
PROTECTED ALLOCATION FLAG	PAT BODY	SCF RECORD WORD 18
CATALOG HASH IF PRMFL	PAT BODY	SCF RECORD WORD 18
PRMFL CATALOG PRESENT FLAG	PAT BODY	SCF RECORD WORD 18
RANDOM FLAG	PAT BODY	SCF RECORD WORD 18
CREATOR NOT USER FLAG	PAT BODY	SCF RECORD WORD 18
FILE IS I-D-S FLAG	PAT BODY	SCF RECORD WORD 18
PURGE FLAG	PAT BODY	SCF RECORD WORD 18

.MPAAF HIERARCHICAL FUNCTION DIAGRAM

	MODULE NUMBER	MODULE NAME
PROCESS TAPE REQUEST	1.2.2	MORTAP

Extract information from the PAT for the appropriate tape file using the PAT obtained in 1.2 MORFIL.
Go to 1.5 Header.

INFORMATION	FROM	TO
TAPE MOUNT FLAG	PAT BODY	WORD 14 OF RECORD
TAPE READY FLAG	PAT BODY	WORD 14 OF RECORD
REEL # = TRUE FILE SERIAL # FLAG	PAT BODY	SCF RECORD WORD 15
REEL NUMBER (IN BCD)	PAT BODY	SCF RECORD WORD 15
ALTERNATE IOM NUMBER	.CRCT1	WORD 18 OF RECORD
ALTERNATE CHANNEL NUMBER	.CRCT1	WORD 18 OF RECORD
ALTERNATE DEVICE NUMBER	SCT	WORD 18 OF RECORD

	MODULE NUMBER	MODULE NAME
RELEASE CORE REQUEST	1.3	RELCOR

Move number of blocks of core released
from .STEMP+10 of the requesting
programs SSA to the SCF record.
Go to routine to fill in areas of header
for slave being serviced.

	MODULE NUMBER	MODULE NAME
RELEASE MASS STORAGE/TAPE REQUEST	1.4	RELFIL

Get the PAT pointer for the released file
from the requesting programs .STEMP+10.
Extract the following information and
move it to the SCF record.
If device not mass storage,
go to 1.4.2 RELTAP.

INFORMATION	FROM	TO
DEVICE TYPE	SCT	SCF RECORD WORD 14
PURGE FLAG	PAT BODY	SCF RECORD WORD 14
DISPOSITION CODE	PAT POINTER	SCF RECORD WORD 14
FILE CODE	PAT POINTER	SCF RECORD WORD 14
CHANNEL USE TIME	PAT BODY	SCF RECORD WORD 15
IOM NUMBER	.CRCT1	SCF RECORD WORD 19
CHANNEL NUMBER	.CRCT1	SCF RECORD WORD 19
DEVICE NUMBER	SCT	SCF RECORD WORD 19

	MODULE NUMBER	MODULE NAME
PROCESS MASS STORAGE RELEASE	1.4.1	RELMAS

Extract information from the PAT for the appropriate mass storage file using the PAT obtained in 1.4 RELFIL.

Go to 1.5 Header.

INFORMATION	FROM	TO
INITIAL SIZE UNAVAILABLE	PAT BODY	SCF RECORD WORD 14
FINAL SIZE UNAVAILABLE	PAT BODY	SCF RECORD WORD 14
INITIAL SIZE IN LLINKS	PAT BODY	SCF RECORD WORD 16
FINAL RELATIVE LLINK	PAT BODY	SCF RECORD WORD 16
CONTINUATION OF FINAL RELATIVE LLINK	PAT BODY	SCF RECORD WORD 17
INITIAL RELATIVE LLINK POSITION	PAT BODY	SCF RECORD WORD 17
MAXIMUM SIZE IN LLINKS	PAT BODY	SCF RECORD WORD 18
PROTECTED ALLOCATION FLAG	PAT BODY	SCF RECORD WORD 19
CATALOG HASH IF PRMFL	PAT BODY	SCF RECORD WORD 19
PRMFL CATALOG PRESENT FLAG	PAT BODY	SCF RECORD WORD 19
RANDOM FLAG	PAT BODY	SCF RECORD WORD 19
PERMANENT FLAG	PAT BODY	SCF RECORD WORD 19
CREATOR NOT USER FLAG	PAT BODY	SCF RECORD WORD 14
FILE IS I-D-S FLAG	PAT BODY	SCF RECORD WORD 19
WRITE FLAG	PAT BODY	SCF RECORD WORD 19
UNPERMITTED ACCESS FLAG	PAT BODY	SCF RECORD WORD 19
PURGE FLAG	PAT BODY	SCF RECORD WORD 19

	MODULE NUMBER	MODULE NAME
PROCESS REMOVABLE MASS STORAGE RELEASE Extract information from the PAT for the appropriate removable mass storage file using the PAT obtained in 1.4 RELFIL. Go to 1.5 Header.	1.4.2	RELMAS

	MODULE NUMBER	MODULE NAME
PROCESS TAPE RELEASE	1.4.3	RELTAP

Extract information from the PAT for the appropriate tape file using the PAT obtained in 1.4 RELFIL.
Go to 1.5 Header.

INFORMATION	FROM	TO
MOUNT INSTRUCTION ISSUED	PAT BODY	SCF RECORD WORD 14
TAPE IN STANDBY	PAT BODY	SCF RECORD WORD 14
NUMBER OF CONNECTS	SCT	SCF RECORD WORD 16
NUMBER OF ERRORS	SCT	SCF RECORD WORD 16
REEL # = TRUE FILE SERIAL # FLAG	PAT BODY	SCF RECORD WORD 17
FILE SERIAL NUMBER	PAT BODY	SCF RECORD WORD 17
ENDING RECORD POSITION	SCT	SCF RECORD WORD 18
DENSITY FLAG	SCT	SCF RECORD WORD 18
ENDING FILE POSITION	SCT	SCF RECORD WORD 18
SERIES 2000 TAPE FLAG	SCT	SCF RECORD WORD 19
ALTERNATE IOM NUMBER	.CRCT1	SCF RECORD WORD 19
ALTERNATE CHANNEL NUMBER	.CRCT1	SCF RECORD WORD 19
ALTERNATE DEVICE NUMBER	SCT	SCF RECORD WORD 19

	MODULE NUMBER	MODULE NAME
FORMAT STANDARD SCF RECORD HEADER	1.5	HEADER

Format Standard Header.
Call .MSCF,2.

SECTION 6

CONSOLE VERBS HFD

.MSCFV
HIERARCHICAL FUNCTION DIAGRAM

	MODULE NUMBER	MODULE NAME
SCF VERBS	001	.MSCFV
SCF RECORD TRACING OFF	1.1	SCFTOF
SCF RECORD TRACING ON	1.2	SCFTON
SCF VERB JOURNALIZING OFF	1.3	SCFVOF
SCF VERB JOURNALIZING ON	1.4	SCFVON
SCF VERB JOURNALIZING STATUS	1.5	SCFVST

SCF RECORD TRACING OFF — 1.1 — SCFTOF

<This section is executed to turn tracing of SCF record types off by number, class, or all tracing.>

Perform COMT 1.1.1.
If record number flag is not zero, then perform TOFFR 1.1.2.
If class number flag is not negative, then perform TOFFC 1.1.3.
If all flag is not zero, then perform TOFFA 1.1.4.
Reset all flags.
Exit.

COMMON TRACE — 1.1.1 — COMT

<This section is performed to analyze the text words and set record, class, or all flags.>

If text equals ALL, then set all flag to one and exit.
If text equals CLASSn, then set class flag to n and exit.
If text equals nnn, then set record number flag to nnn and exit.
Issue error message.
Exit.

TRACE OFF BY RECORD — 1.1.2 — TOFFR

Take SCF record number out of record number flag and isolate by class and number.
Get the trace map word for the class.
Set the bit for the record number off.

	MODULE NUMBER	MODULE NAME
Store the trace map word for the class. Exit.		
TRACE OFF BY CLASS Take SCF class number of class ;flag. Get the trace map word for the class. Turn all the bits off. Store the trace map word for the class. Exit.	1.1.3	TOFFC
ALL TRACING OFF Get all trace map words in order. Turn all the bits off. Store all trace map words in order. Exit.	1.1.4	TOFFA
SCF RECORD TRACING ON <This section is executed to turn tracing of SCF record types on by number, class, or all tracing.> Perform COMT 1.1.1. If record number flag is not zero, then perform TONR 1.2.1. If class number flag is not negative, then perform TONC 1.2.2. If all flag is not zero, then perform TONA 1.2.3. Reset all flags. Exit.	1.2	SCFTON
TRACE ON BY RECORD Take SCF record number out of record number flag and isolate by class and number. Get the trace map word for the class. Set the bit for the record number on. Store the trace map word for the class. Exit.	1.2.1	TONR

.MSCFV
HIERARCHICAL FUNCTION DIAGRAM

	MODULE NUMBER	MODULE NAME
TRACE ON BY CLASS Take SCF class number out of class flag. Get the trace map word for the class. Turn all the bits on. Store the trace map word for the class. Exit.	1.2.2	TONC
ALL TRACING ON Get all trace map words in order. Turn all the bits on. Store all trace map words in order. Exit.	1.2.3	TONA
SCF VERB JOURNALIZING OFF <This section is executed to turn journalization of console verbs off by name, class, or all journalization.> Perform COMV 1.3.1. If verb flag is not zero, then perform VOFFN 1.3.2. If class flag is not negative, then perform VOFFC 1.3.3. If all flag is not zero, then perform VOFFA 1.3.4. Reset all flags. Exit.	1.3	SCFVOF
COMMON VERB <This section is performed to analyze the text words and set verb, class, or all flag.> If text equals ALL, then set all flag to one and exit. If text equals CLASSn, then set class flag to n and exit. If text equals a valid verb name, then set verb flag to class and number and exit. Issue error message. Exit.	1.3.1	COMV

11/21/77

HIERARCHICAL FUNCTION DIAGRAM

	MODULE NUMBER	MODULE NAME
VERB OFF BY NAME	1.3.2	VOFFN
Get the verb flag and isolate class and number. Get VMAP word for the class. If the disposition bit is off, issue can't change message exit. Else turn the bit off and store the VMAP word. Exit.		
VERB OFF BY CLASS	1.3.3	VOFFC
Get VMAP word for this class. If disposition bit is off, issue can't change message. Exit. Else, turn all the bits off. Store the VMAP word. Exit.		
ALL VERBS OFF	1.3.4	VOFFA
Get VMAP words in turn. If disposition bit is off, issue can't change message. Exit. Else turn the bits off. Store the VMAP words in turn. Exit.		
SCF VERB JOURNALIZING ON	1.4	SCFVON
<This section is executed to turn journalization of console verbs on by name, class, or all journalization.> Perform COMV 1.3.1. If verb flag is not zero, then perform VONN 1.4.5. If class flag is not negative, then perform VONC 1.4.2. If all flag is not zero, then perform VONA 1.4.3. Reset all flags. Exit.		

.MSCFV
HIERARCHICAL FUNCTION DIAGRAM

	MODULE NUMBER	MODULE NAME
VERB ON BY NAME	1.4.1	VONN

```
   Get the verb flag and isolate by class
      and number.
   Get VMAP word for the class.
      If disposition bit is off,
      issue can't change message.
      Exit.
      else turn the bit on.
      store the VMAP word.
      Exit.
```

VERB ON BY CLASS	1.4.2	VONC

```
   Get VMAP word for this class.
   If disposition bit is off,
      issue can't change message.
      Exit.
      Else turn all the bits on.
      Store the VMAP word.
      Exit.
```

ALL VERBS ON	1.4.3	VONA

```
   Get VMAP words in turn.
   If disposition bit is off,
      issue can't change message.
      Exit.
      Else turn the bits on.
      Store the VMAP words in turn.
      Exit.
```

SCF VERB JOURNALIZING STATUS	1.5	SCFVST

```
      <This section is executed to check the
      status of console verb journalization by
      name, class, and all journalization.>
   Perform COMV 1.3.1.
   If verb flag is not zero,
      then perform VSTN 1.5.1.
   If class flag is not negative,
      then perform VSTC 1.5.2.
   If all flag is not zero,
      then perform VSTA 1.5.3.
   Reset all flags.
   Exit.
```

11/21/77

.MSCFV
HIERARCHICAL FUNCTION DIAGRAM

	MODULE NUMBER	MODULE NAME
VERB STATUS BY NAME Get the verb flag and isolate by class and number. Get VMAP word for the class. If disposition bit is off, and verb vit is off, issue name with IN. If disposition bit is off and verb bit is on, issue name with AN. If disposition bit is on, and verb bit is off, issue name with IC If disposition bit is on, and verb bit is on, issue name with AC Exit.	1.5.1	VSTN
VERB STATUS BY CLASS Get the class flag. Get the VMAP word for the class. Respectively perform VSTN for each valid verb in the class. Exit.	1.5.2	VSTC
VERB STATUS ALL Respectively perform VSTC for each of the 10 classes of verbs. Exit.	1.5.3	VSTA

MODULE NUMBER - 773

MODULE SYMBOL - .MSCFV

MODULE NAME - SCF Verb Journalization and Tracing

MODULE FUNCTION

The primary functions of this module are as follows:

- Setting trace off for SCF record types
- Setting trace on for SCF record types
- Setting journalization of console verbs on
- Setting journalization of console verbs off
- Checking the journalization status of console verbs

The module will be responsible for capturing the text of SCFTOF, SCFTON, SCFVOF, SCFVON, and SCFVST, checking for legal parameters and either performing the desired functions or reporting to the console any detected errors in operator input.

MODULE REFERENCED BY - 215 .MPOP7

MODULES REFERENCED - None

MODULE NUMBER - 1.1

MODULE SYMBOL - SCFTOF

MODULE NAME - SCF Record Tracing Off

MODULE FUNCTION

Entry into this module is via a GEPOP Queue entry. The responsibility of this module is to turn off the appropriate trace map (TMAP) bit based on the contents of the text of the operator command.

This module first transfers to COMT to get the record, class, and all flags set and then based on the contents of the three flags, it will transfer to:

TOFFR if record flag is not zero,
TOFFC if class flag is not negative, or
TOFFA if all flag is not zero.

It will then reset all the flags to a neutral state and exit.

INTERNAL REFERENCES

MODULES REFERENCED - COMT(1.1.1) TOFFR(1.1.2)
TOFFC(1.1.3) TOFFA(1.1.4)

MODULE REFERENCED BY - None

EXTERNAL REFERENCES

MODULES REFERENCED - None

MODULE REFERENCED BY - .MPOPM

SYSTEM GATES REFERENCED - None

COMMUNICATIONS REGIONS REFERENCED - .CRACF

MODULE NUMBER - 1.1.1

MODULE SYMBOL - COMT

MODULE NAME - Common Trace

MODULE FUNCTION

This module is called via an internal TSX and will first access the text at .STEMP+8,5. The text will then be compared to characters "ALL" and if the comparison is good, will set all flags equal to one and transfer to exit. It will next compare the text for the string "CLASS" and if the comparison is good, it will take the numeric diget following, store it in class flag, and transfer to exit. It will next check for a text of three numeric characters and if true, will store the number in record flag and transfer to exit. Otherwise, it will issue a message to the console and exit.

INTERNAL REFERENCES

MODULES REFERENCED - None

MODULE REFERENCED BY - SCFTOF(1.1) SCFTON(1.2)

EXTERNAL REFERENCES - None

MODULE NUMBER - 1.1.2

MODULE SYMBOL - TOFFR

MODULE NAME - Trace Off by Record

MODULE FUNCTION

This module is entered via an internal TSX. The first thing it does is isolate the value in record flag by class and number. It will then set up a mask word of all one bits except the bit corresponding to the record number and then AND the mask and the appropriate class word of TMAP via .CRACF+16.

INTERNAL REFERENCES

MODULES REFERENCED - None

MODULE REFERENCED BY - SCFTOF(1.1)

EXTERNAL REFERENCES - None

SYSTEM GATES REFERENCED - .CRACF

COMMUNICATIONS REGIONS REFERENCED - .CRACF

CALLING SEQUENCE - TSX1 TOFFR,$

MODULE NUMBER - 1.1.3

MODULE SYMBOL - TOFFC

MODULE NAME - Trace Off by Class

MODULE FUNCTION

This module is entered via an internal TSX. The first thing it does is to build a mask word of zeros for ANDing into TMAP+n, where n is the value in class flag. It then ANDs the mask and the appropriate class word of TMAP via .CRACF+16.

INTERNAL REFERENCES

MODULES REFERENCED - None

MODULE REFERENCED BY - SCFTOF(1.1)

EXTERNAL REFERENCES - None

SYSTEM GATES REFERENCED - .CRACF

COMMUNICATIONS REGIONS REFERENCED - .CRACF

CALLING SEQUENCE - TSX1 TOFFC,$

MODULE NUMBER - 1.1.4

MODULE SYMBOL - TOFFA

MODULE NAME - All Tracing Off

MODULE FUNCTION

This module is entered via an internal TSX. The first thing it does is to build a mask word of zeros for ANDing into all words of TMAP. It then ANDs all words of TMAP via .CRACF+16.

INTERNAL REFERENCES

MODULES REFERENCED - None

MODULE REFERENCED BY - SCFTOF(1.1)

EXTERNAL REFERENCES - None

SYSTEM GATES REFERENCED - .CRACF

COMMUNICATIONS REGIONS REFERENCED - .CRACF

CALLING SEQUENCE - TSX1 TOFFA,$

MODULE NUMBER - 1.2

MODULE SYMBOL - SCFTON

MODULE NAME - SCF Record Tracing ON

MODULE FUNCTION

Entry into this module is via a GPOP Queue entry. The responsibility of this module is to turn on the appropriate trace map (TMAP) bits based on the contents of the text of the operator command.

The module first transfers to COMT to get the record, class, and all flags set and then based on the contents of the three flsgs, it will transfer to:

TONR if record flag is not zero,
TONC if class flag is not negative, or
TONA if all flag is not zero.

It will then reset all the flags to a neutral state and exit.

INTERNAL REFERENCES

MODULES REFERENCED - COMT(1.1.1) TONR(1.2.1)
TONC(1.2.2) TONA(1.2.3)

MODULE REFERENCED BY - None

EXTERNAL REFERENCES

MODULES REFERENCED - None

MODULE REFERENCED BY - .MPOPM

SYSTEM GATES REFERENCED - None

COMMUNICATIONS REGIONS REFERENCED - .CRACF

MODULE NUMBER - 1.2.1

MODULE SYMBOL - TONR

MODULE NAME

Trace On by Record

MODULE FUNCTION

This module is entered via an internal TSX. The first thing it does is isolate the value in record flag by class and number. It will then set up a mask word of all zero bits except the bit corresponding to the record being turned on and then OR's the mask and the appropriate class word of TMAP via .CRACF+16.

INTERNAL REFERENCES

MODULES REFERENCED - None

MODULE REFERENCED BY - SCFTON(1.2)

EXTERNAL REFERENCES - None

SYSTEM GATES REFERENCED - .CRACF

COMMUNICATIONS REGIONS REFERENCED - .CRACF

CALLING SEQUENCE - TSX1 TONR,$

MODULE NUMBER - 1.2.2

MODULE SYMBOL - TONC

MODULE NAME - Trace On by Class

MODULE FUNCTION

This module is entered via an internal TSX. The first thing it does is to build a mask word of one bit for the appropriate number of bits for ORing into TMAP+n, where n is the value in class flag. It then OR's the mask and the appropriate class word of TMAP via .CRACF+16.

INTERNAL REFERENCES

MODULES REFERENCED - None

MODULE REFERENCED BY - SCFTON(1.2)

EXTERNAL REFERENCES - None

SYSTEM GATES REFERENCED - .CRACF

COMMUNICATIONS REGIONS REFERENCED - .CRACF

CALLING SEQUENCE - TSX1 TONC,$

MODULE NUMBER - 1.2.3

MODULE SYMBOL - TONA

MODULE NAME - All Tracing ON

MODULE FUNCTION

This module is entered via an internal TSX. The first thing it does is to build a mask word of one bit for ORing into the first word of TMAP. It then OR's the first word of TMAP via .CRACF+16. This is then done for each of the other words of TMAP.

INTERNAL REFERENCES

MODULES REFERENCED - None

MODULE REFERENCED BY - SCFTON(1.2)

EXTERNAL REFERENCES - None

SYSTEM GATES REFERENCED - .CRACF

COMMUNICATIONS REGIONS REFERENCED - .CRACF

CALLING SEQUENCE - TSX1 TONA,$

MODULE NUMBER - 1.3

MODULE SYMBOL - SCFVOF

MODULE NAME - SCF Verb Journalizing OFF

MODULE FUNCTION

Entry into this module is via a GEPOP Queue entry. The responsibility of this module is to turn off the appropriate verb map (VMAP) bits based on the contents of the text of the operator command.

This module first transfers to COMV to get the verb, class and all flags set and then based on the contents of the three flags, it will transfer to:

VOFFN if verb flag is not zero or
VOFFC is class flag is not negative or
VOFFA if all flag is not zero.

It will then reset all the flags to a neutral state and exit.

INTERNAL REFERENCES

MODULES REFERENCED - COMV(1.3.1) VOFFN(1.3.2)
VOFFC(1.3.3) VOFFA(1.3.4)

MODULE REFERENCED BY - None

EXTERNAL REFERENCES

MODULES REFERENCED - None

MODULE REFERENCED BY - .MPOPM

SYSTEM GATES REFERENCED - .CRACF

COMMUNICATIONS REGIONS REFERENCED - .CRACF

MODULE NUMBER - 1.3.1

MODULE SYMBOL - COMV

MODULE NAME - Common Verbs

MODULE FUNCTION

This module is called via an internal TSX and it will first access the text at .STEMP+8,5. The text will then be compared to "ALL" and if true will set the all flag equal to one and transfer to exit. It will next compare the text to "CLASS" and if true take the numbric diget following, store it in class flag, and transfer to exit. It will next take the first six characters of text and search against the operator verb name table and if a comparison is true it will put the class and verb number in verb flag, and transfer to exit. Otherwise it will issue an error message to the console and exit.

INTERNAL REFERENCES

MODULES REFERENCED - None

MODULE REFERENCED BY - SCFVOF(1.3) SCFVON(1.4) SCFVST(1.5)

EXTERNAL REFERENCES - None

SYSTEM GATES REFERENCED - .CRPOQ

COMMUNICATIONS REGIONS REFERENCED - .CRPOQ

CALLING SEQUENCE - ESX1 COMV,$

MODULE NUMBER - 1.3.2

MODULE SYMBOL - VOFFN

MODULE NAME - Verb OFF by Name

MODULE FUNCTION

This module is entered via an internal TSX. The first thing it does is isolate the value in name flag by searching the verb name table (VNTAB) and on finding a match obtaining the verb class and number. If no match is found an error message is issued to the console. It will then get the class word of VMAP via .CARCF+15 and check the disposition bit. If bit 0 is 0, a message is issued to the console that the disposition can't be changed. Otherwise, a mask is created for ANDing the proper bit off and the word is stored back into VMAP.

INTERNAL REFERENCES

MODULES REFERENCED - None

MODULE REFERENCED BY - SCFVOF(1.3)

EXTERNAL REFERENCES - None

SYSTEM GATES REFERENCED - .CRACF

COMMUNICATIONS REGIONS REFERENCED - .CRACF

CALLING SEQUENCE - TSX1 VOFFN,$

MODULE NUMBER - 1.3.3

MODULE SYMBOL - VOFFC

MODULE NAME - Verb OFF by Class

MODULE FUNCTION

This module is entered via an internal TSX. The first thing it does is to get the value in class flag and then get the appropriate word of VMAP via .CRACF+15. It then checks the disposition bit. If bit 0 is 0, a message is issued to the console that the disposition can't be changed. Otherwise all the bits in the word are masked off and the word is stored in VMAP via .CRACF+15.

INTERNAL REFERENCES

MODULES REFERENCED - None

MODULE REFERENCED BY - SCFVOF(1.3)

EXTERNAL REFERENCES - None

SYSTEM GATES REFERENCED - .CRACF

COMMUNCIATIONS REGIONS REFERENCED - .CRACF

CALLING SEQUENCE - TSX1 VOFFC,$

MODULE NUMBER - 1.3.4

MODULE SYMBOL - VOFFA

MODULE NAME - All Verbs OFF

MODULE FUNCTION

This module is entered via an internal TSX. Each word of VMAP is processed in turn. Starting with the first word of VMAP obtained via .CRACF+15 the disposition bit is checked. If 0 is 0 a message is issued to the console that the disposition can't be changed otherwise the bits are ANDed off and the word is stored back into VMAP via .CRACF+15. Then the next word is processed.

INTERNAL REFERENCES

MODULES REFERENCED - None

MODULE REFERENCED BY - SCFVOF(1.3)

EXTERNAL REFERENCES - None

SYSTEM GATES REFERENCED - .CRACF

COMMUNICATIONS REGIONS REFERENCED - .CRACF

CALLING SEQUENCE - TSX1 VOFFA,$

MODULE NUMBER - 1.4

MODULE SYMBOL - SCFVON

MODULE NAME - SCF Verb Journalizing ON

MODULE FUNCTION

Entry into this module is via a GEPOP Queue entry. The responsibility of this module is to turn on the appropriate verb map (VMAP) bit based on the contents of the text of the operator command.

This module first transfers to COMV to get the verb, class and all flags set and then based on the contents of the three flags, it will transfer to:

VONN if verb flag is not zero or,
VONC is class flag is not negative, or
VONA if all flags is not zero.

It will then reset all the flags to a neutral state and exit.

INTERNAL REFERENCES

MODULES REFERENCED - COMV(1.3.1) VONN(1.4.1)
VONC(1.4.2) VONA(1.4.3)

MODULE REFERENCED BY - None

EXTERNAL REFERENCES

MODULES REFERENCED - None

MODULE REFERENCED BY - .MPOPM

SYSTEM GATES REFERENCED - .CRACF

COMMUNICATIONS REGIONS REFERENCED - .CRACF

MODULE NUMBER - 1.4.1

MODULE SYMBOL - VONN

MODULE NAME - Verb ON by Name

MODULE FUNCTION

This module is entered via an internal TSX. The first thing it does is isolate the value in name flag by searching the verb name table (VNTAB) and on finding a match, obtaining the verb class and number. If no match is found an error message is issued to the console. It will then get the class word of VMAP via .CRACF+15 and check the disposition bit. If 0 is 0, a message is issued to the console that the disposition can't be changed. Otherwise a mask is created for ORing the proper bit on and the word is stored back into VMAP.

INTERNAL REFERENCES

MODULES REFERENCED - None

MODULE REFERENCED BY - SCFVON(1.4)

EXTERNAL REFERENCES - None

SYSTEM GATES REFERENCED - None

COMMUNICATIONS REGIONS REFERENCED - .CRACF

CALLING SEQUENCE - TSX1 VONN,$

MODULE NUMBER - 1.4.2

MODULE SYMBOL - VONC

MODULE NAME - Verb On by Class

MODULE FUNCTION

This module is entered via an internal TSX. The first thing it does is to get the value in class flag and then get the appropriate word of VMAP via .CRACF+15. It then checks the disposition bit. If bit 0 is 0, a message is issued to the console that the disposition can't be changed. Otherwise, the bits in the word are masked on for the number of vlaid verbs in this class and the word is stored in VMAP via .CRACF+15.

INTERNAL REFERENCES

MODULES REFERENCED - None

MODULES REFERENCED BY - SCFVON(1.4)

EXTERNAL REFERENCES - None

SYSTEM GATES REFERENCED - .CRACF

COMMUNICATIONS REGIONS REFERENCED - .CRACF

CALLING SEQUENCE - TSX1 VONC,$

MODULE NUMBER - 1.4.3

MODULE SYMBOL - VONA

MODULE NAME - All Verbs ON

MODULE FUNCTION

This module is entered via an internal TSX. Each word of VMAP is processed in turn. Starting with the first word of VMAP obtained via .CRACF+15, the disposition bit is checked. If 0 is 0, a message is issued to the console that the disposition can't be changed otherwise the bits are ORed on for the proper number of verbs in the class and the word is stored back into VMAP via .CRACF+15. Then the next word is processed.

INTERNAL REFERENCES

MODULES REFERENCED - None

MODULE REFERENCED BY - SCFVON(1.4)

EXTERNAL REFERENCES - None

SYSTEM GATES REFERENCED - .CRACF

COMMUNICATIONS REGIONS REFERENCED - .CRACF

CALLING SEQUENCE - TSX1 VONA,$

MODULE NUMBER - 1.5

MODULE SYMBOL - SCFVST

MODULE NAME - SCF Verb Journalizing Status

MODULE FUNCTION

Entry into this module is via a GEPOP Queue entry. The responsibility of this module is to report the status of verb journalization based on the contents of the text of the operator command.

This module first transfers to COMV to get the verb, class and all flags set and then based on the contents of the three flags, it will transfer to:

VSTN if verb flag is not zero,
VSTC if class flag is not negative, or
VSTA if all flag is not zero.

It will then reset all the flags to a neutral state and exit.

INTERNAL REFERENCES

MODULES REFERENCED - COMV(1.3.1) VSTN(1.5.1)
VSTC(1.5.2) VSTA(1.5.3)

MODULE REFERENCED BY - None

EXTERNAL REFERENCES

MODULES REFERENCED - None

MODULE REFERENCED BY - .MPOPM

SYSTEM GATES REFERENCED - .CRACF

COMMUNICATIONS REGIONS REFERENCED - .CRACF

MODULE NUMBER - 1.5.1

MODULE SYMBOL - VSTN

MODULE NAME - Verb Status by Name

MODULE FUNCTION

This module is entered via an internal TSX. The first thing it does is isolate the value in name flag by searching the verb name table (VNTAB) and on finding a match, obtaining the verb class and number. If no match is found an error message is issued to the console. It will then get the class word of VMAP via .CRACF+15 and check the disposition and the number of the verb bit. If the disposition bit is off and verb bit is off, a message is issued to the console containing the name with an IN following. Likewise for AN, IC, and AC. It then exits.

INTERNAL REFERENCES

MODULES REFERENCED - None

MODULE REFERENCED BY - SCFVST(1.5)

EXTERNAL REFERENCES - None

SYSTEM GATES REFERENCED - .CRACF

COMMUNICATIONS REGIONS REFERENCED - .CRACF

CALLING SEQUENCE - TSX1 VSTN,$

MODULE NUMBER - 1.5.2

MODULE SYMBOL - VSTC

MODULE NAME - Verb Status by Class

MODULE FUNCTION

This module is entered via an internal TSX. The first thing it does is to get the value in class flag and then get the appropriate word of VMAP via .CRACF+15. It also finds the entry in VNTAB that corresponds to the first verb of the class. It then repetitively calls VSTN for each verb in the class and exits.

INTERNAL REFERENCES

MODULES REFERENCED - None

MODULE REFERENCED BY - SCFVST(1.5)

EXTERNAL REFERENCES - None

SYSTEM GATES REFERENCED - .CRACF

COMMUNICATIONS REGIONS REFERENCED - .CRACF

CALLING SEQUENCE - TSX1 VSTC,$

MODULE NUMBER - 1.5.3

MODULE SYMBOL - VSTA

MODULE NAME - Verb Status All

MODULE FUNCTION

This module is entered via an internal TSX. VSTC is performed for each of the 10 classes of verbs and then exits.

INTERNAL REFERENCES

MODULES REFERENCED - None

MODULE REFERENCED BY - SCFVST(1.5)

EXTERNAL REFERENCES - None

SYSTEM GATES REFERENCED - .CRACF

COMMUNICATIONS REGIONS REFERENCED - .CRACF

CALLING SEQUENCE - TSX1 VSTA,$ MODULE NUMBER - 1.3

MODULE SYMBOL - CALSCF (.MSCF,2)

MODULE NAME - SCF Processing and SCF Trace Queue Management

MODULE FUNCTION

Entry into this module is made via an external call from a GCOS element. Upon entry, the related modules expect to find the address of an SCF record in index register 1. Checks to determine if a particular SCF record is to be written or traced are made by the calling element by using the .SCFCK macro. CALSCF also checks if the record is to be recorded, traced or both. If the .SCFCK macro is not utilized by the caller to check the recording status of a given record prior to issuing a call to .MSCF, the record will only be written if the recording status indicates such. The caller cannot force a record to be recorded.

Responsibilities of this module are:

- Saving references to the SCF Record to be processed
- Checking the recording and trace status of the record type
- SCF output buffer management
- Completion of the SCF Header
- Output control and error detection
- SCF trace entry management
- Generation of System Counters record upon detection of job start or job termination records

Output control and error detection are accomplished by a courtesy call routine which executes as .MPOPM. Output is started by a CALSCF module that executes under the program number of the caller. The SCF output buffers are located in .MPOPM and are identified by the symbol ACC.AB.

It is possible that a callers SCF record will not be placed in the SCF file. This event will occur under the following conditions:

1. SCF is not configured

2. SCF is not initialized

3. Operator requested SCF file exchange is in progress and a partially filled buffer is being written.

4. Disk SCF is configured and one SCF file is purging and an exchange has been initiated (software/operator) and the operator has to 'wait' for the purge to complete before effecting the exchange.

A count of lost logical records is maintained within the module. This count is recorded within System Counters record, type 504. The caller is not informed that the record has been lost.

Code within the module is inhibited. The initial sequence within the module performs preliminary tests to determine if an SCF file is configured and available. Failure of this test sequence will result in a return to the calling element. If the previous tests were successful, the record is a candidate for placement in the SCF buffer (RECORD 1.3.3). Upon return, checks are made to determine if a system counters record is required. Upon detection of job start or job end/delete, this record is formatted by SYSCTR 1.3.2 and RECORD is performed to complete processing.

Completion of processing results by invoking the EXIT 1.6 module.

INTERNAL REFERENCES

MODULES REFERENCED - EXIT 1.3.1 SYSCTR 1.3.2 MAPCK 1.3.5
RECORD 1.3.3 TRACE 1.3.4
MEXIT 1.3.2

MODULE REFERENCED BY - None

EXTERNAL REFERENCES

MODULES REFERENCED - See .MSCF 001

MODULE REFERENCED BY - See .MSCF 001

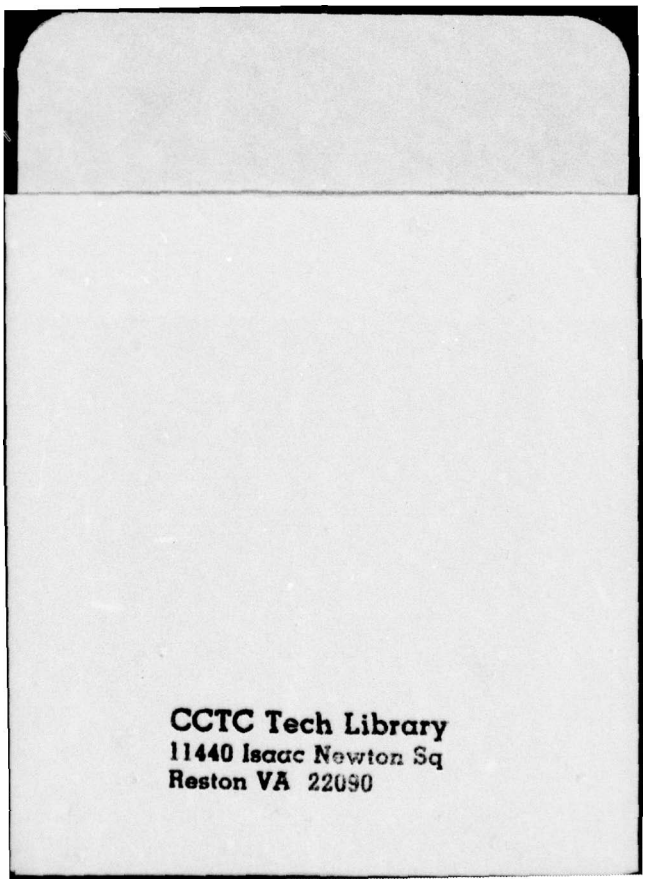
SYSTEM GATES REFERENCED - .CRACF
.CRSTC

COMMUNICATIONS REGIONS REFERENCED

.CRACF .CRSTC .CRETR

CALLING SEQUENCE

```
EAX1 (address of callers SCF record)
.CALL .MSCF,2
```


CCTC Tech Library
11440 Isaac Newton Sq
Reston VA 22090